



Gestión automática para la solicitud de pedidos a través de aplicaciones de mensajería en establecimientos comerciales como Gastrobares.

Diego Alejandro Rodriguez Gamboa

Código: 10892127871

Sandra Yamile Rodriguez Gamboa

Código: 10892124608

Universidad Antonio Nariño

Programa Especialización Ingeniería de Software

Facultad de Ingeniería de Software

Bogotá, Colombia

2022

Gestión automática para la solicitud de pedidos a través de aplicaciones de mensajería en establecimientos comerciales como Gastrobares.

Diego Alejandro Rodriguez Gamboa - Sandra Yamile Rodriguez Gamboa

Proyecto de grado presentado como requisito parcial para optar al título de:

Especialista Ingeniería de Software

Director (a):

Msc. Dianalin Neme Prada

Codirector (a):

Msc. Iván Rodrigo Romero

Universidad Antonio Nariño

Programa Especialización Ingeniería de Software

Facultad de Ingeniería de Software

Bogotá, Colombia

2022

NOTA DE ACEPTACIÓN

El trabajo de grado titulado

Gestión automática para la solicitud de pedidos a través de aplicaciones de mensajería en establecimientos comerciales como Gastrobares.

Cumple con los requisitos para optar

Al título de Especialista de Ingeniería de Software.

Firma del Tutor

Firma Jurado

Firma Jurado

Bogotá, 24 05 2022.

Contenido

Listado de Tablas	6
Listado de Figuras	7
Resumen	9
Abstract	10
Introducción	11
1. Título	12
2. Formulación y Descripción del problema	13
2.1 Descripción del Problema	13
2.2 Formulación del Problema	13
3. Objetivo General	15
4. Objetivos Específicos	15
5. Marco de Referencia	16
5.1 Estado del Arte	16
5.2 Impacto	18
5.2.1 Impacto Social	18
5.2.2 Impacto tecnológico	19
5.2.3 Impacto económico	19
5.3 Componente de Innovación	19
5.4 Marco teórico	20
6. Metodología	23
7. Proceso de Software	29
7.1 Requerimientos Funcionales	29
7.1.1 Pedidos RF1	29
7.1.2 Mensajería Automática RF2	30
7.2 Requerimientos no Funcionales FNF0	31
7.2.1 Restricciones	31
7.3 Diseño y Arquitectura	32
7.3.1 Diagrama de Despliegue	32
7.3.2 Caso de uso arquitecturalmente relevante	34
7.3.3 Diagrama de Secuencia	37
7.3.4 Diagrama de Clases	38
7.3.5 Arquitectura de alto nivel	39
8. Construcción	40
9. Pruebas	49

9.1 Pruebas de Proceso	49
9.2 Pruebas de Producto.....	52
9.3 Pruebas de Seguridad.....	57
10. Instalación y Configuración.....	59
10.1 Instalación de las librerías Phyton	59
10.2 Instalación de NGROK.....	60
10.3 Conexión con Twilio.....	63
11. Conclusiones	67
12. Anexo.....	69
12.1 Análisis de mercado y estudio de necesidades.....	69
12.2 Implementación de políticas de seguridad.....	69
12.3 Análisis de activos, riesgos, controles y efectividad.....	69
12.4 Documento arquitectura SAD.....	69
13. Referencias.....	70

Lista de Tablas

Tabla 1 <i>Empresas que ofrecen servicios de Chatbot</i>	17
Tabla 2 <i>Artefacto Scrum</i>	24
Tabla 3 <i>Roles</i>	26
Tabla 4 <i>Requerimientos funcionales Pedidos</i>	29
Tabla 5 <i>Requerimientos funcionales mensajería automática</i>	30
Tabla 6 <i>Requerimientos no funcionales</i>	31
Tabla 7 <i>Criterios de aceptación</i>	53

Lista de Figuras

Ilustración 1 <i>Diseño preliminar HLD</i>	21
Ilustración 2 <i>Ilustración Scrumban</i>	23
Ilustración 3 <i>Artefactos Scrumban</i>	25
Ilustración 4 <i>Cronograma</i>	28
Ilustración 5 <i>Diagrama de despliegue</i>	34
Ilustración 6 <i>Diagrama de casos de uso</i>	36
Ilustración 7 <i>Diagrama de secuencia</i>	37
Ilustración 8 <i>Diagrama de clases</i>	38
Ilustración 9 <i>Arquitectura de alto nivel</i>	39
Ilustración 10 <i>Patrón arquitectura Broken</i>	40
Ilustración 11 <i>Estructura de una red neuronal</i>	42
Ilustración 12 <i>Instalación librería TensorFlow</i>	43
Ilustración 13 <i>Código fuente entrenamiento red neuronal</i>	44
Ilustración 14 <i>Construcción código QR</i>	45
Ilustración 15 <i>Construcción diccionario de datos</i>	46
Ilustración 16 <i>Imágenes multimedia</i>	47
Ilustración 17 <i>Código phyton</i>	48
Ilustración 18 <i>Medición código a través de SonarCloud</i>	50
Ilustración 19 <i>Tiempo de respuesta del chatBot</i>	51
Ilustración 20 <i>Código QR</i>	54
Ilustración 21 <i>Inicio conversacion en WhastApp</i>	55
Ilustración 22 <i>Imágenes flujo de conversación con el chatbot.</i>	56

Ilustración 23 <i>Top 10 SonarQube</i>	57
Ilustración 24 <i>Análisis vulnerabilidades de funcionalidad y seguridad</i>	58
Ilustración 25 <i>Instalación librerías Phyton</i>	59
Ilustración 26 <i>URL para descarga Ngrok</i>	60
Ilustración 27 <i>Folder de descargas</i>	61
Ilustración 28 <i>Ejecución Ngrok</i>	62
Ilustración 29 <i>Creación cuenta Twilio</i>	63
Ilustración 30 <i>Configuración cuenta Twilio</i>	64
Ilustración 31 <i>Configuración Sandbox</i>	65
Ilustración 32 <i>Conversación con ChatBot</i>	66

Resumen

Hoy en día el componente diferenciador entre las empresas es el servicio y el valor que este implica en la respuesta inmediata y conectividad constante para presentar estos servicios y productos a sus clientes de manera oportuna y ágil para así mantenerlos siempre satisfechos. Es ahí donde la tecnología empieza a jugar un papel fundamental para prestación de servicios más eficientes y eficaces, con respuestas inmediatas y concretas. Esta tecnología cada día nos ofrece más y mejores oportunidades de desarrollo implementando técnicas como el manejo de inteligencia artificial, entrenamiento de redes neuronales, haciendo uso de herramientas de comunicación más utilizadas a nivel mundial y haciendo una combinación de estas tres podemos ofrecer a las empresas un desarrollo que le permita en este caso a los Gastrobares poder tener una comunicación eficiente, oportuna, ágil y eficaz con sus clientes a la hora de realizar un pedido o solicitar la cuenta de su consumo.

Palabras Claves: Inteligencia artificial, redes neuronales, reconocimiento de voz, NLTK Lenguaje natural, Twilio, Ngrok, menú, gastrobar, comensal, mesa.

Abstract

Today the differentiator component between companies is the service and the value that this implies in the immediate response to present these services and products to their clients in a timely and agile manner in order to always keep them satisfied. This is where technology begins to play a very important role in providing more efficient and effective services, with immediate and concrete responses. This technology every day offers us more and better opportunities for development by implementing techniques such as the use of artificial intelligence, training of neural networks, using the most widely used communication tools worldwide and by combining these three we can offer companies that in this case allows to the Gastrobars to have efficient, timely, agile and effective communication with their customers when doing order or requesting their billing.

Keywords: Artificial intelligence, neural networks, speech recognition, NLTK Natural Language, Twilio, Ngrok, menu, gastro-bar.

Introducción

Los chatbots hoy en día pueden proporcionar experiencias útiles, divertidas y únicas a los usuarios convirtiéndolos en clientes habituales, proporcionan una tecnología extremadamente versátil ya que pueden seguir patrones de preguntas y respuestas que imita el vaivén de una conversación humana real. Como resultado de este proceso, los empleados podrán disponer de más tiempo y energía para centrarse en consultas de gran complejidad y así mejorar el servicio al cliente convirtiéndolo en una experiencia única.

En la actualidad las empresas y cualquier negocio en general tienen que invertir sus esfuerzos en la atención inmediata, eficiente y eficaz a sus clientes, para eso buscan diferentes estrategias de marketing y herramientas que ayuden a mejorar la experiencia de usuario y que también puedan estar disponible 7*24.

Así mismo durante los últimos 2 años la humanidad ha experimentado eventos que han cambiado la forma de relacionarnos unos a otros, una pandemia mundial ha hecho que hábitos comunes entre nosotros cambien, es así como las empresas y/o negocios han tenido que implementar medidas de bioseguridad para proteger a sus empleados y clientes. Los negocios más afectados en este escenario son los que tienen que ver con la gastronomía y la diversión.

En este orden de ideas, este proyecto busca cubrir las necesidades de negocios como los gastrobares suministrándoles una herramienta que puedan usar de manera fácil y confiable, usando tecnología de vanguardia, que deje huella en sus clientes con un servicio rápido, eficaz, moderno y hasta divertido y que a su vez pueda seguir protegiendo a sus empleados y clientes con las medidas de bioseguridad que se han tenido que implementar, como es el distanciamiento físico.

1. Título

Gestión automática para la solicitud de pedidos a través de aplicaciones de mensajería en establecimientos comerciales como Gastrobares.

2. Formulación y Descripción del problema

2.1 Descripción del Problema

Se han identificado dos categorías del problema en el que actualmente se ven involucrados los establecimientos comerciales Gastrobares.

La primera tiene que ver con la situación de salud pública que actualmente se vive a nivel mundial la cual ha hecho cambiar sustancialmente la manera de interactuar socialmente entre individuos, a implementar protocolos de bioseguridad como el distanciamiento físico entre las personas para buscar minimizar el contagio en sitios cerrados y públicos como por ejemplo en los establecimientos comerciales como Gastrobares de manera que las personas sientan confianza y se genere tranquilidad a todos en este tipo de espacios.

La segunda está enfocada en la atención al cliente y en los tiempos de respuesta en la atención a la mesa en el momento de solicitar un pedido y/o la cuenta a pagar por parte de los meseros, ya bien sea por la gran concurrencia de comensales en estos establecimientos comerciales o la falta de personal como meseros causando una percepción de mala atención y mal servicio del establecimiento comercial.

2.2 Formulación del Problema

Este proyecto permitiría minimizar el contacto generado en un establecimiento comercial como Gastrobares entre el mesero y el comensal al permitir a un cliente realizar el pedido y solicitud de la cuenta directamente a la cocina y a la caja respectivamente a través de una comunicación en línea (Chatbot) y usando aplicaciones de mensajería como el WhatsApp.

¿Qué medios tecnológicos se pueden utilizar para mejorar la calidad del servicio en establecimientos comerciales como Gastrobares minimizando el contacto físico entre el mesero y el cliente y a su vez generar satisfacción en los tiempos de espera a su atención?

3. Objetivo General

Gestionar de manera automática la solicitud de pedidos y valor de la cuenta directamente a la cocina y caja respectivamente de los comensales en los establecimientos comerciales como Gastrobares, usando aplicaciones de mensajería con respuestas automáticas implementando en ellas Inteligencia Artificial.

4. Objetivos Específicos

- Disminuir el tiempo de respuesta en un 80% en el proceso de la atención inmediata al comensal a partir del momento en que se inicie el escaneo del código QR en el sitio.
- Entregar una implementación en la cual el 60% del flujo conversacional con el Chatbot sea con Inteligencia Artificial.
- Afianzar que los tiempos de respuesta en la interacción con el Chatbot no supere los 5 segundos a pesar de la implementación de componentes de Inteligencia Artificial.
- Humanizar los flujos conversaciones del Chatbot a través de componentes multimedia (audio/imágenes) en un 70% de los flujos.
- Generar la entrega de mensajes automáticos a los clientes correspondientes a las promociones que ofrecen los gastrobares.

5. Marco de Referencia

5.1 Estado del Arte

Hoy en día la tecnología como parte de la innovación continua y buscando estar siempre a la vanguardia del mercado en diferentes modelos de negocio es implementada con mayor frecuencia en negocios donde aún es escasa. La pandemia ha dado un impulso y ha puesto en evidencia las necesidades de un mercado que ha sido en parte relegado por las carencias propias de este, principalmente afectando establecimientos comerciales como Gastrobares por todo el tema de bioseguridad que hoy en día debe implementarse en este tipo de comercios.

Estas innovaciones se han dado para cubrir parte de estas falencias de ese mercado y han dado como resultado el robustecimiento de aplicaciones móviles de domicilio y otros modelos de negocio como son:

Aplicaciones móviles: Domicilios.com, Rappi e incluso algunos restaurantes cuentan con sus propias aplicaciones de domicilios, sin embargo, son especializadas y usadas únicamente en domicilios, incluso aplicaciones donde se da la opción de recoger el pedido en el local.

Dentro de la investigación realizada se encontró dentro del repositorio de la Universidad UAN una aplicación desarrollada por Nixon Rolando Ramos Delgado y Cesar Augusto Leal Reyes llamada “Aplicación para restaurantes de corrientazo, LUNCHAPP” que consiste en “Diseñar una herramienta App para usuario y Web para restaurantes que permita realizar el cargue de menú a los restaurantes afiliados por la Web y autogestión de los usuarios por la App, al momento de hacer sus pedidos en los restaurantes que ofrecen corrientazo para los empleados de oficina y transeúntes en la calle 72 entre las avenidas 14 y 5 de la ciudad de Bogotá.” Leal, C. (2020, Noviembre 11) Aplicación para restaurantes de corrientazo LUNCHAPP [Especialización Ingeniería de Software]. Universidad Antonio Nariño, Bogotá.

Modelos de negocio: MUY es un restaurante que carece de meseros. El local le permite al cliente hacer la orden de su pedido a través de una pantalla instalada en la pared de sus locales, dicha orden es generada y enviada a las cocinas del restaurante desde donde llaman al cliente por su nombre y le entregan el pedido de su plato después de realizar el pago de este.

El comercio y en general cualquier modelo de negocio debe incluir dentro de sus ofertas de servicios tecnología innovadora que ayude a mejorar sus ingresos y su servicio aumentando sus oportunidades comerciales.

Tecnologías como el Chatbot permiten hoy en día automatizar las conversaciones con clientes ofreciendo mayor efectividad y eficiencia en sus comunicaciones con disponibilidad de 24 horas los 7 días de la semana. Estos son usados en diferentes estructuras, modelos de negocios y empresas las cuales empiezan a incluir este tipo de tecnología dentro de sus ofertas.

De acuerdo con la investigación realizada hemos encontrado algunos ejemplos de empresas y proyectos que ofrecen una solución inteligente para automatizar conversiones con sus clientes para cualquier modelo de negocio, los cuales relacionamos en la siguiente tabla:

Tabla 1

Empresas que ofrecen servicios de Chatbot

Empresa	Modelo de Negocio	Líneas de negocio	Fuentes
SnatchBot	SnatchBot ofrece diferentes plantillas ya determinadas de conversaciones automatizadas en diferentes campos comerciales.	Empresas enfocadas en las áreas de Servicio Cliente , Tienda en línea , Hostelería , Negocio , Sanitaria , Banca , Marketing/Herramientas , Educación	https://snatchbot.me/botstore/template
Cliengo	Cliengo brinda a sus empresas la oportunidad iniciar una conversación	Empresas que tengan un dominio Web para sus negocios.	https://www.cliengo.com/en/chatbot

	24/7 con todos los visitantes de su sitio Web mientras captura sus datos de contacto.		
Aivo	Aivo a través de su solución Agentbot ofrece a sus clientes la oportunidad de construir sus propios flujos conversaciones.	Empresas dirigidas a la investigación de datos.	https://www.aivo.co/chatbot-agentbot
Botifica	Botifica a través de su solución Yami ofrece una solución de Chatbot para restaurantes basado en Messenger.	Soluciones a la medida, han implementado Chatbot especialmente para ayudar a empresar con su marketing digital en áreas como Ecommerce, eventos, Webinars, Educación, dentistas, políticos.	https://botifica.com/

Fuente: (Diseño propio)

5.2 Impacto

Se han identificado tres clases de impactos en los cuales este proyecto puede afectar al sector de Gastrobares tanto para los clientes como para los dueños de estos establecimientos comerciales:

5.2.1 Impacto Social

Un proyecto de estas características impactaría significativamente al recurso humano ya que se disminuiría la interacción entre clientes y meseros de los diferentes establecimientos. El hecho de implementar los diferentes protocolos de bioseguridad decretados por el Gobierno Nacional ha generado un cambio estructural respecto al servicio prestado en estos negocios por lo que este proyecto puede ayudar a que estos protocolos se puedan implementar haciendo uso de esta tecnología y así mantener una conversación con el Chatbot.

5.2.2 Impacto tecnológico

Este proyecto usará la herramienta de comunicación más utilizada por diferentes usuarios en el mundo como es el WhatsApp y así no habrá la necesidad de descargas adicionales de aplicaciones. Además, el restaurante deberá ser consciente de adquirir equipos tecnológicos que permitan que esta solución tenga éxito ya que gracias a la interacción entre el cliente y el Chatbot esta información será mostrada tanto al chef como el gerente a través de dispositivos periféricos.

5.2.3 Impacto económico

El impacto económico que un proyecto como este se verá afectado la forma como los costos administrativos serán reemplazados en costos operativos: Debido, por ejemplo, a la disminución del personal, ya que la interacción entre clientes y meseros es mínima esta será sustituida por los costos que generará el servicio de mensajería de Twilio que es mínimo en comparación con la carga prestacional y/o administrativa del establecimiento comercial.

5.3 Componente de Innovación

Este proyecto pretende no solo leer el Código QR sino generar la solicitud de pedido en línea a través de una conversación con un Chatbot y a su vez solicitar la cuenta a pagar, ofreciendo al cliente respuestas inmediatas en cuestión de segundos para que la espera entre la llegada al establecimiento y la solicitud del pedido sea mínima. En medio de este flujo conversacional el Chatbot podrá generar mensajes automáticos de atención o la disponibilidad de agregar, quitar o cancelar el pedido realizado.

El Chatbot usará librerías para implementar Inteligencia Artificial como Tensor Flow, NLTK que permitirá utilizar patrones de comportamiento y reconocimiento de voz en el que el Chatbot estará en la capacidad de interpretar de manera correcta. Al tratarse de un lugar de esparcimiento y en un ambiente muy informal el robot podrá humanizarse incluyendo respuestas

y mensajes de audio en algunos momentos de la conversación, ofreciendo al cliente una experiencia agradable, entretenida, e incluso hasta divertida.

Este proyecto pretende utilizar herramientas que existen actualmente en el mercado y que la gran mayoría de las personas utilizan hoy en día como es el WhatsApp, esto hace que no sea necesario que tanto el comensal como dueño del Gastrobar descarguen aplicaciones extras en sus dispositivos móviles.

5.4 Marco teórico

En la actualidad, se puede observar como la tecnología avanza muy rápido y que las empresas siempre estarán buscando la manera de brindar y mejorar un servicio de calidad al usuario final para volverse más competitivas y a la vanguardia del negocio. Es por esto por lo que una de las nuevas tecnologías más buscadas, usadas y que tienen un componente de innovación continua son las que se están trabajando por asistentes virtuales (Chatbot).

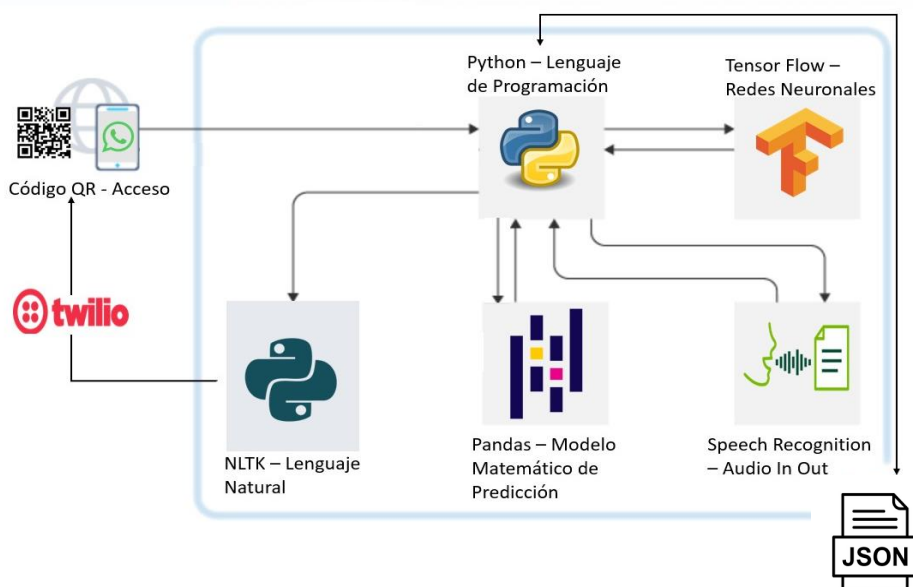
La automatización en el área de TI no es ajena al sector de establecimientos comerciales como los Gastrobares, buscando la optimización de tiempos y mano de obra obteniendo resultados en el mejoramiento continuo del servicio, el sector de Gastrobares quiere empezar a brindar un mejor servicio de calidad e innovación, automatizando algunos de sus procesos para garantizar servicios de calidad y usando herramientas sencillas, simples y accesibles a cualquier tipo de usuarios.

La solución propuesta para este proyecto es implementar un Chatbot con una interfaz vía WhatsApp para mejorar esta gestión, empleando librerías como Tensor Flow (Redes Neuronales), Pandas (Modelo Matemático de Predicción), Speech recognition (Audio in out) y NLTK (Lenguaje natural) del lenguaje de programación libre de Python, todas estas librerías

conectadas a través del manejo con Json para el intercambio de datos necesarios en el flujo conversacional el cual será implementado utilizando el API de comunicación que ofrece Twilio.

Ilustración 1

Diseño preliminar HLD



Fuente: (Diseño propio)

Chatbot: Es definido como un programa que simula conversaciones humanas (escritas o incluso habladas) mostrado al usuario mediante una interfaz, conectada a un número determinado de algoritmos y datos API, lo que permite proporcionar servicios e información bajo demanda.

Python: Lenguaje de programación con librerías que permiten generar flujos dinámicos, llamativos y creativos (sonido, imágenes) lo cual hará que sea más amena y clara la interacción con el usuario final.

Tensor Flow: Es una biblioteca de Python que se utiliza para construir y entrenar redes neuronales y así obtener patrones de comportamiento que se utilizan para aplicaciones con inteligencia artificial.

Speech Recognition: Audio in out. Librería de Python que permite el manejo de reconocimiento de voz para programar las respuestas del Chatbot con mensajería de voz.

Pandas: Modelo Matemático de predicción. Librería de Python que es usada para la manipulación y análisis de datos, ofreciendo estructuras de datos y operaciones para manipular tablas numéricas desde ficheros en formatos como CSV, Excel y base de datos.

NLTK: Lenguaje Natural. Conjunto de librerías de procesamiento del lenguaje natural, considerada una subdisciplina de la inteligencia artificial, ocupada de la formulación de mecanismos eficaces para la comunicación entre personas y máquinas por medio de lenguajes naturales.

Twilio: Tecnología que nos permitirá conectar los flujos realizados en el lenguaje de programación con el envío en lenguaje natural al usuario final vía WhatsApp.

Ngrok: Herramienta con la que se puede establecer un sitio público con el localhost de la máquina de forma rápida y fácil.

Json: Es un formato ligero de intercambio de datos, de fácil lectura y escritura para los usuarios. También nos ayuda a realizar fácil análisis.

6. Metodología

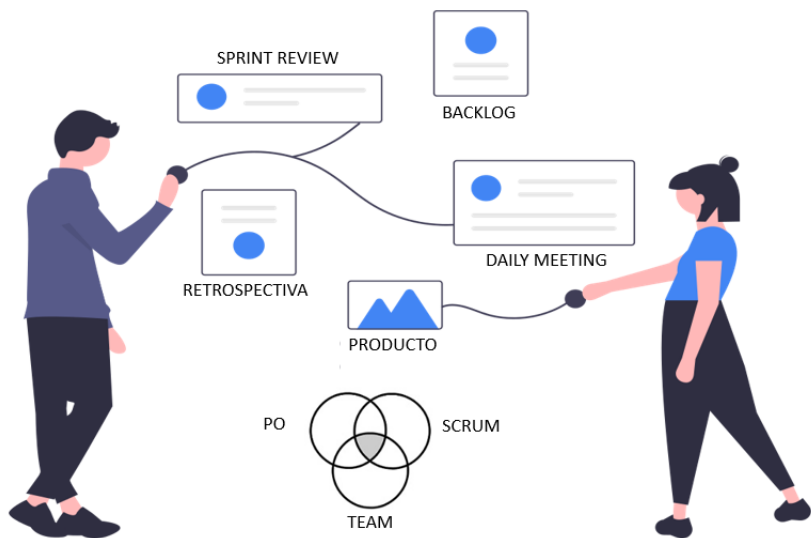
METODOLOGÍA SCRUMBAN

Para la ejecución de este proyecto, se propone emplear una Metodología Ágil bajo un marco de trabajo totalmente colaborativo llamado SCRUMBAN el cual apunta a formar equipo de trabajo entre las personas relacionadas como del equipo de desarrollo y construcción del Chatbot. En esta metodología empleamos equipos multidisciplinarios, auto organizados y auto gestionados; con capacidad de respuesta ágil; en los diferentes temas de construcción y soporte.

Los equipos de trabajo totalmente emergentes, motivados y orientados a entregar resultados en corto tiempo.

Ilustración 2

Ilustración Scrumban



Fuente: Diseño propio

Este equipo de trabajo operara bajo el esquema ágil fallar rápido, cuanto más cerca esté una falla al punto donde se introdujo, más información tendrá el colaborador para corregir esta falla y, por consiguiente, lograr el cierre del entregable.

Apoyado con los artefactos de Scrum, se realiza toda la implementación de los componentes de software según sprint planning del Backlog dado por el Product Owner. Quien ha alimentado las historias de usuario que se deben llevar a cabo, dejando los criterios de aceptación establecidos en cada una de estas, para realizar por el equipo de trabajo que de manera evolutiva atenderá las solicitudes del backlog.

Tabla 2

Artefacto Scrum

Artefacto	Descripción
Product Backlog	Lista de requisitos de usuario, que a partir de la visión inicial del producto crece y evoluciona durante el desarrollo.
Sprint backlog	Lista de los trabajos que debe realizar el equipo durante el sprint para generar el incremento previsto.
HU	Historia de Usuario, con la descripción de requerimiento a realizar.
Sprint	Nombre que recibe cada iteración de desarrollo. Es el núcleo central que genera el pulso de avance por tiempos
Incremento	Resultado de cada sprint.
Daily	Reunión diaria breve, de no más de 15 minutos, en la que el equipo sincroniza el trabajo y establece el plan para las 24 horas siguientes.
Tableros Kanban	Con la lista de tareas ya armada, estamos en condiciones de crear el tablero. donde básicamente se divide en 3 columnas: pendientes, en curso y terminadas.
Definition of Done	Lista de chequeo que permite validar que una HU completa.
Definition of Ready	Lista de chequeo que permite validar un conjunto de HU están listas para ser aprobadas por el usuario o stakeholder.

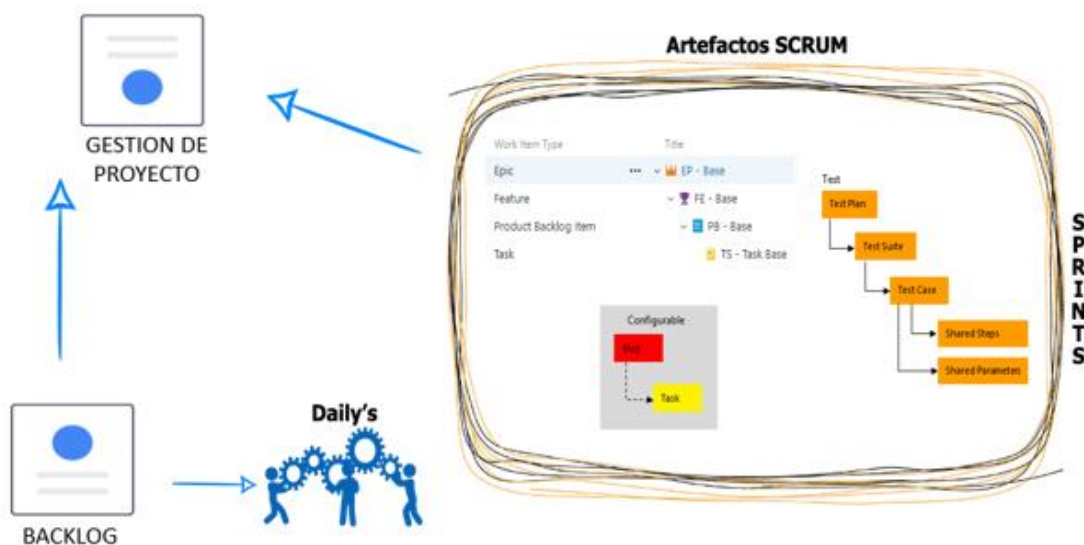
Fuente: Diseño propio

La metodología es aplicada durante toda la ejecución del proyecto, desde su fase inicial a su fase final, gracias a las diferentes herramientas de gestión del mercado se tiene la posibilidad

de gestionar proyectos ágiles, definiendo cada uno de los artefactos scrum, esto combinado con herramientas de comunicación permite tener equipos completamente conectados, facilitando una comunicación asertiva sin importar su ubicación física.

Ilustración 3

Artefactos Scrumban



Fuente: Diseño propio

Tabla 3*Roles*

Rol	Descripción del Rol
<p>Scrum Master (SM) Diego Rodriguez</p>	<p>Ingeniero de sistemas o carreras afines. Asegura que Scrum sea entendido y adoptado en los equipos de trabajo. Asegura que el Equipo Scrum trabaja ajustándose a la teoría, prácticas y reglas de Scrum sabiendo aplicar la metodología a cada entorno de manera práctica.</p> <ul style="list-style-type: none"> a) Asegurar que el equipo sea multifuncional y eficiente. b) Proteger al equipo de distracciones y aspectos externos al proyecto. c) Detectar, monitorear y facilitar la remoción de los impedimentos que puedan surgir con respecto al proyecto y a la metodología acordada. d) Conducir al equipo hacia el logro de sus actividades. e) Realizar balanceo de cargas conforme a necesidad del negocio. f) Asegurar la cooperación y comunicación dentro del equipo de trabajo. g) Garantizar el uso de los procesos, procedimientos, metodologías y herramientas; definidos por el cliente y acordados con la fábrica (enfoque AGIL o TRADICIONAL). h) Asegurar que todos los integrantes del equipo asistan a tiempo a las reuniones diarias (máximo 15 minutos), en donde el equipo se debe concentrar en responder tres preguntas: ¿Qué hice? (verificar cumplimiento de compromisos) ¿Qué voy a hacer? (generar nuevos compromisos) ¿Qué problemas se presentaron? (identificar impedimentos, pero no se resuelven en esta reunión). i) Asistir a las reuniones de planeación y seguimiento, junto con el responsable(s) del entregable (Cliente), de tal manera que se pueda obtener un feedback temprano y poder determinar qué acciones puede implementar en el siguiente entregable. j) Acordar planes de trabajo efectuando balanceo de cargas de tal manera que se pueda cumplir con las necesidades del cliente.
<p>Product Owner (PO) Sandra Rodriguez</p>	<p>Ingeniero de sistemas o carreras afines. Representante de los clientes que usan el software. Se focaliza en la parte de negocio y él es responsable del ROI del proyecto (entregar un valor superior al dinero invertido).</p> <ul style="list-style-type: none"> a) Visión del Proyecto o servicio. b) Velar por el cumplimiento del servicio. c) Participar activamente de las reuniones de solicitudes, seguimiento o cualquier aspecto que involucre ajustes. d) Administrar, monitorear y coordinar los equipos de trabajo de la fábrica. e) Velar por los resultados de cada una de las iteraciones. f) Servir de interlocutor con el cliente.

	<p>g) Aportar ideas innovadoras para satisfacer los requerimientos, de tal forma que sea un valor agregado del servicio.</p> <p>h) Participar de la revisión de seguimiento, junto con los miembros del equipo y el responsable asignado por el cliente, para obtener el respectivo Feedback y velar por que este sea aplicado.</p> <p>i) Coordinar con cliente el agendamiento de las actividades de seguimiento a nivel Gerencial.</p> <p>j) Garantizar la calidad del servicio.</p> <p>k) Presentar periódicamente al cliente informes emergentes de Gestión, con los indicadores establecidos para el control del servicio. Se sugiere parciales (semanales) y definitivos (mensuales).</p> <p>l) Capacidad de toma de decisiones para las estimaciones y conocimiento a nivel técnico y de negocio del cliente.</p> <p>m) Servir entre interlocutor entre los líderes del cliente y los colaboradores de la fábrica, cuando se estime conveniente.</p>
Arquitecto Diego Rodriguez	<p>Ingeniero de sistemas o carreras afines, con especialización. Realiza diseños:</p> <ul style="list-style-type: none"> ✓ Componentes ✓ Interfaces ✓ Integraciones ✓ Módulos del sistema ✓ Asegura el cumplimiento de los requerimientos no funcionales
Analista de requerimientos Sandra Rodriguez	<p>Realiza el levantamiento y documentación de:</p> <ul style="list-style-type: none"> ✓ Documentación de requerimientos ✓ Elaboración y diagramación de casos de uso ✓ Elaboración modelos conceptuales y diagramas de procesos.
Analista de pruebas Sandra Rodriguez	<p>Ingeniero de sistemas o carreras afines. Encargado del aseguramiento de calidad de los componentes de software construido, realizara pruebas funcionales sobre las historias de usuario y de aplicar dentro del proyecto la ejecución pruebas no funcionales.</p>
Analista de Desarrollo Diego Rodriguez	<p>Ingeniero de Sistemas, con habilidades técnicas necesarias para realizar construcción de componentes de software, aplicando los lineamientos y buenas prácticas dadas por la compañía para garantizar calidad en cada uno de los entregables.</p>
Analista de Despliegues Diego Rodriguez	<p>Analistas de despliegue con conocimiento en las plataformas y herramientas, conocimiento del negocio y conocimiento en diferentes tipos de despliegue. Este equipo está conformado por los colaboradores necesarios para la realización las actividades, y deberá en lo posible; ser lo más autoorganizados. Los Analistas de despliegue, son los principales colaboradores en las labores del servicio, y como tal son responsables, entre otras, de:</p> <ul style="list-style-type: none"> a) Mantener y controlar el plan detallado de ejecución, b) Documentar los hallazgos, c) Reportar cualquier tipo de novedad, de cada una de las plataformas en pruebas,

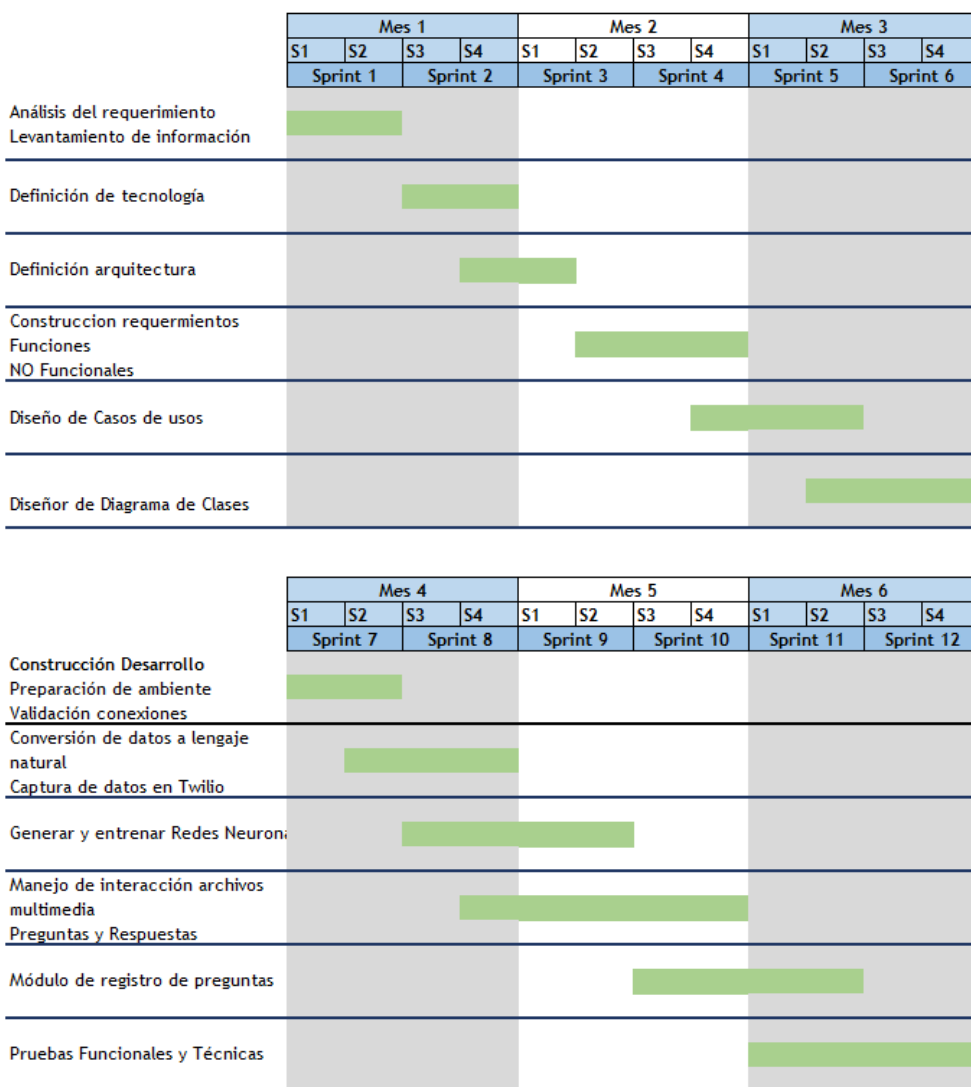
- | | |
|--|---|
| | d) Generar las evidencias correspondientes, a los procesos de calidad ejecutados. |
| | e) Cumplir el proceso definido con el cliente. |

Fuente: Diseño propio

La definición del cronograma es realizada por sprints con una duración de 2 semanas cada uno, en cada sprint se elaboró una parte importante del proyecto involucrando los roles necesarios para cumplir con la entrega.

Ilustración 4

Cronograma



Fuente: Diseño propio

7. Proceso de Software

7.1 Requerimientos Funcionales

A partir de la priorización de requerimientos se identificaron 3 categorías principales asociadas a la funcionalidad de estos: pedidos, solicitar cuenta, mensajería automática.

A continuación, se analizan los requerimientos más importantes de cada categoría:

7.1.1 Pedidos RF1

Tabla 4

Requerimientos funcionales Pedidos

	Justificación	Alcanzable	Medible	Prioridad
RF101 El sistema debe <u>permitir</u> al cliente realizar pedidos según las opciones ofrecidas por el chatbot.	Este requerimiento hace parte del Core del negocio ya que permite al cliente la solicitud de pedidos según la disponibilidad y opciones presentadas por el chatbot.	Este requerimiento se podrá cumplir siguiendo la disponibilidad completa de los productos ofrecidos.	Se revisará a partir de solicitud realizada por el cliente en un momento vs la disponibilidad actual del producto.	Alta
RF102 El sistema debe permitir al administrador ver las solicitudes de cuenta de las diferentes mesas para tener una atención a la cliente inmediata.	Este requerimiento hace parte del Core del negocio y permite la opción al administrador ver todas las solicitudes de cuentas para poder distribuir de manera adecuada el cobro de estas.	Este requerimiento se podrá cumplir adaptando una vista de administrador donde él pueda generar la distribución adecuada del cobro de cuentas.	Se medirá teniendo en cuenta el número de solicitudes de cuenta vs el tiempo de respuesta de atención a la mesa.	Alta

Fuente: Diseño Propio

7.1.2 Mensajería Automática RF2

Tabla 5*Requerimientos funcionales mensajería automática*

	Justificación	Alcanzable	Medible	Prioridad
RF201 El sistema debe generar de manera automática alertas que tienen que ver con la satisfacción del servicio.	Este requerimiento hace parte del Core del negocio y permite al administrador y chef tener una retroalimentación del cliente y así poder generar un proceso de mejora continua.	Este requerimiento se podrá cumplir generando un job que se active y envíe mensajes aleatorios de satisfacción del servicio.	Se medirá teniendo en cuenta las respuestas generadas por el cliente.	Alta
RF202 El sistema debe generar de manera automática alertas que tienen que ver con descuentos y marketing del negocio.	Este requerimiento hace parte del Core del negocio y permite la opción al administrador generar ofertas sobre sus productos y también hacer propaganda a su negocio.	Este requerimiento se podrá cumplir generando un job que se active y envíe mensajes aleatorios de ofertas y propaganda.	Se medirá teniendo en cuenta el número de vistas y los pedidos realizados de los productos en oferta.	Alta

Fuente: Diseño Propio

7.2 Requerimientos no Funcionales FNF0

Tabla 6

Requerimientos no funcionales

	Justificación	Criterio de aceptación	Restricciones	Medible	Prioridad
RNF001 El sistema debe generar respuesta a la solicitud realizada por el cliente en menos de 2 segundos.	Es importante que el usuario vea una respuesta a cada solicitud realizada al chatbot en tiempos cortos para garantizar una óptima experiencia.	Al realizar una solicitud al chatbot de cualquier tipo el contenido de la respuesta se debe realizar en menos de 2 segundos	La IA debe garantizar mostrar la opción adecuada	La presentación de las opciones se cumple en los límites de tiempo establecidos	Alta
RF002 El envío automático de mensajería el sistema NO debe enviar datos que vulneren la seguridad de clientes (números de tarjetas, nombres, entre otros).	Es importante que el usuario no vea que se están utilizando datos personales para el envío de mensajes automáticos.	Al realizar un envío de mensajería automática este no debe tener información sensible del cliente.	N/A	Los filtros de seguridad que generan alertas por envío de información privada	Alta

Fuente: Diseño Propio

7.2.1 Restricciones

- Costos del servicio de mensajería que debe ser manejada por el servicio de Twilio.
- Costos para infraestructura, alojamiento de la aplicación y base datos ya sea en la nube o onpremise.

- Tecnología obsoleta de clientes, con celulares de gama baja que no permitan el escaneo de un código QR.
- El recurso humano para la implementación de este proyecto debe cumplir una curva de aprendizaje no mayor de 2 meses en el lenguaje y herramientas que son necesarias para la implementación de este en producción.
- Restricciones que a nivel legal pueda tener WhatsApp en Colombia para compartir información del restaurante.

7.3 Diseño y Arquitectura

En esta sección del documento se realizará la identificación de algunos diagramas de UML (Lenguaje Unificado de Modelado) para entender mejor de manera visual el enfoque dado para el proyecto.

7.3.1 Diagrama de Despliegue

En el siguiente diagrama de despliegue se puede visualizar los procesadores/nodos/dispositivos del hardware que se usara en el sistema, así como los enlaces de comunicación entre ellos.

Se implementará el servidor de lógica empleando un servidor de Azure remoto, que, a través de configuración, ofrecerá una capa de servicios para el acceso a la funcionalidad del chatbot, En este se podrán dividir las funciones que tienen que ver con el negocio (solicitud de pedidos, solicitud de cuentas y alertas automáticas), de la lógica de inteligencia artificial (redes neuronales) que hacen parte de la esencia del chatbot.

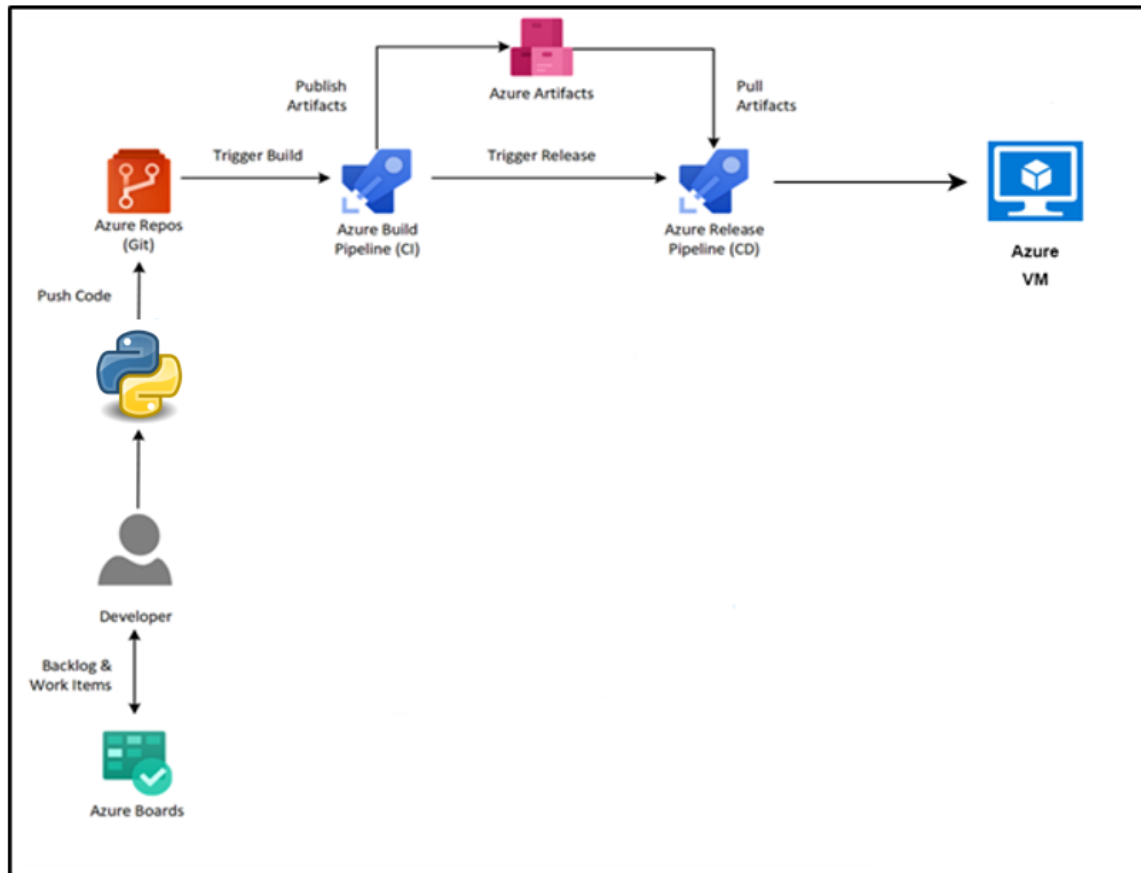
El servidor contara un sistema operativo de Windows server 2012 ya que permite una configuración sencilla tipo Plug and play que permite la rápida configuración de servicios que hayan sido desarrollados en tecnología Microsoft.

Adicionalmente se presenta la conexión con un sistema externo, de mensajería el cual permitirá que los mensajes viajen del procesamiento de lenguaje natural a el cliente vía WhatsApp

Finalmente, la persistencia será manejada mediante el intercambio de datos a través de Json, esta conexión se realizará de manera automática empleando bibliotecas de conexión en Python.

Ilustración 5

Diagrama de despliegue



Fuente: Diseño propio

7.3.2 Caso de uso arquitecturalmente relevante

Se requiere realizar la automatización del proceso de pedidos en un Gastrobar conectando el código QR no al menú sino a un chatbot con el que se pueda interactuar en el mismo momento del escaneo y pueda no solo ver el menú sino realizar las siguientes acciones:

1. Flujo de pedidos: Una vez se escanea el código QR, se enviará el menú en imagen vía WhatsApp y el usuario podrá seleccionar una opción de pedido tanto bebidas como comida, una vez el usuario seleccione la opción esta será enviada por WhatsApp al

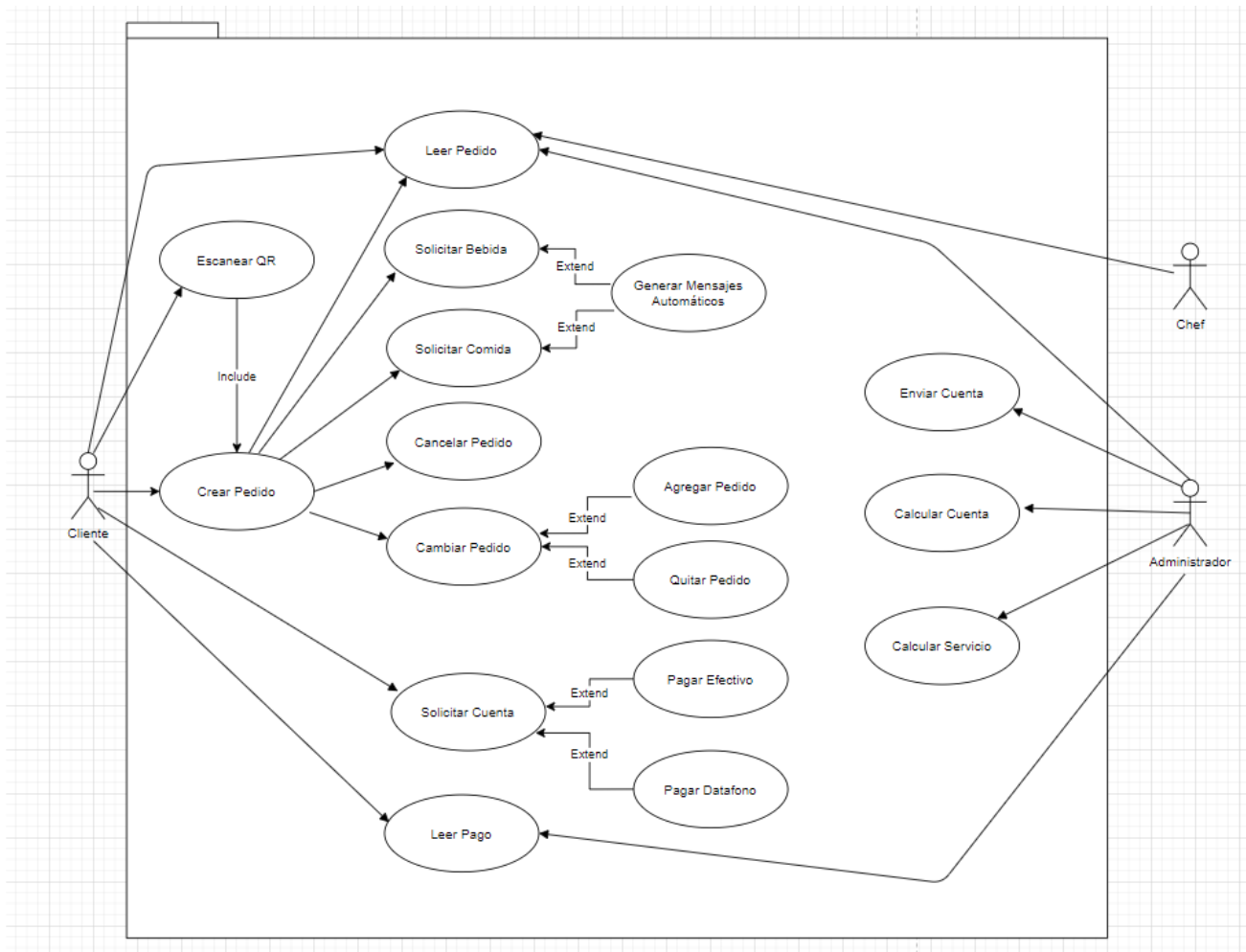
administrador. El cliente también tendrá la opción de cambiar su pedido o cancelarlo según requiera.

El administrador/chef leerá los pedidos y empezará su alistamiento para llevarlos a la mesa, el tiempo de entrega de estos pedidos será realizado bajo la complejidad de preparación de cada pedido, así se garantiza el orden de solicitud las solicitudes.

2. Flujo espera: Después de realizar el pedido y pasados algunos minutos el Chatbot podrá empezar a enviar mensajes automáticos con líneas de servicios adicionales que el cliente pueda necesitar.
3. Flujo cuenta: Cuando el cliente termine de comer, podrá pedir la cuenta por WhatsApp y el administrador calculara el valor de la cuenta junto con el valor del servicio y responderá a través del Chatbot el valor a pagar y le preguntara la forma pago ya sea efectivo o datafono y si es efectivo preguntara con que billete pagara para recibir el cambio, el administrador/chef leerá la respuesta para enviar a la persona con el datafono y el cambio según sea el caso.

Ilustración 6

Diagrama de casos de uso



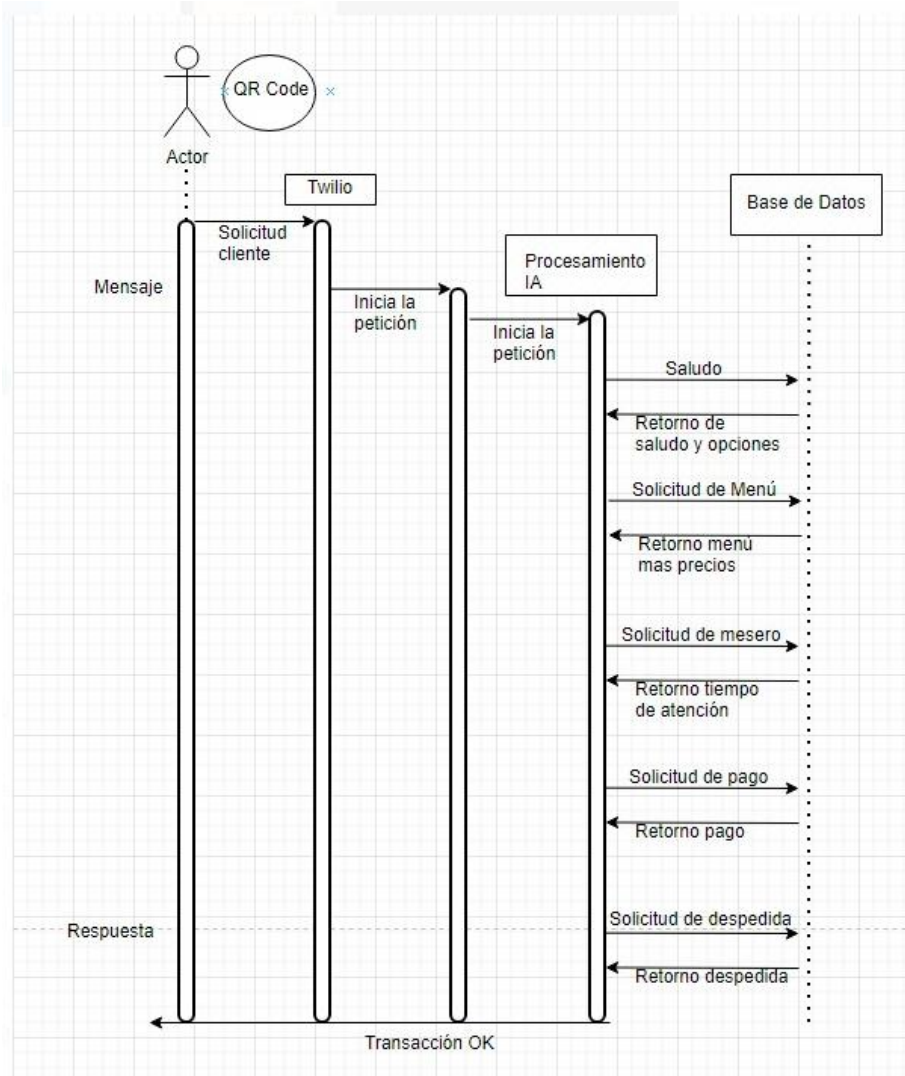
Fuente: Diseño propio

7.3.3 Diagrama de Secuencia

El siguiente diagrama de secuencia muestra los flujos por los que el sistema deberá pasar para cumplir con las especificaciones de los requerimientos,

Ilustración 7

Diagrama de secuencia



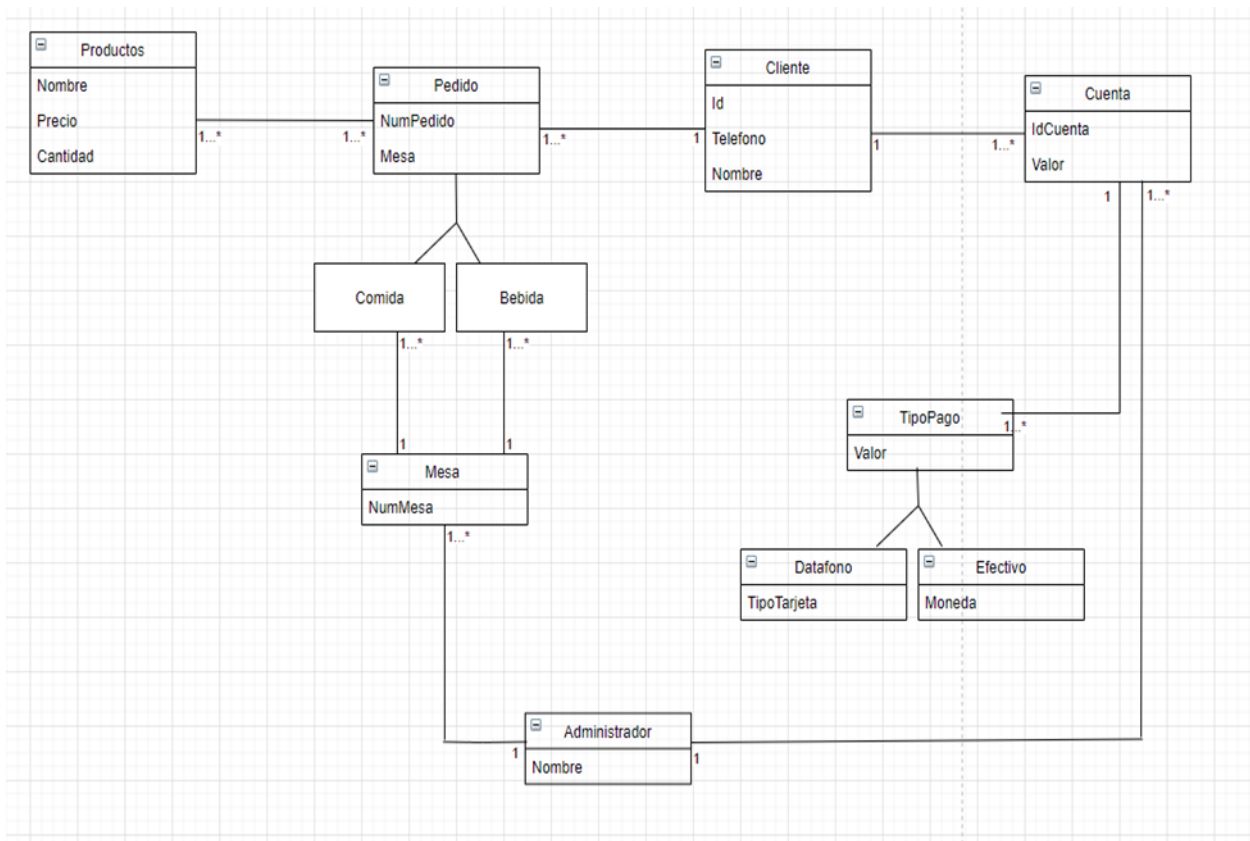
Fuente: Diseño propio

7.3.4 Diagrama de Clases

En el siguiente diagrama de clases se identifican los elementos que componen el sistema desde un punto de vista estático. Se han identificado 7 clases diferentes con sus respectivos atributos, así mismo 2 subclases, realizando la respectiva relación entre ellas.

Ilustración 8

Diagrama de clases



Fuente: Diseño propio

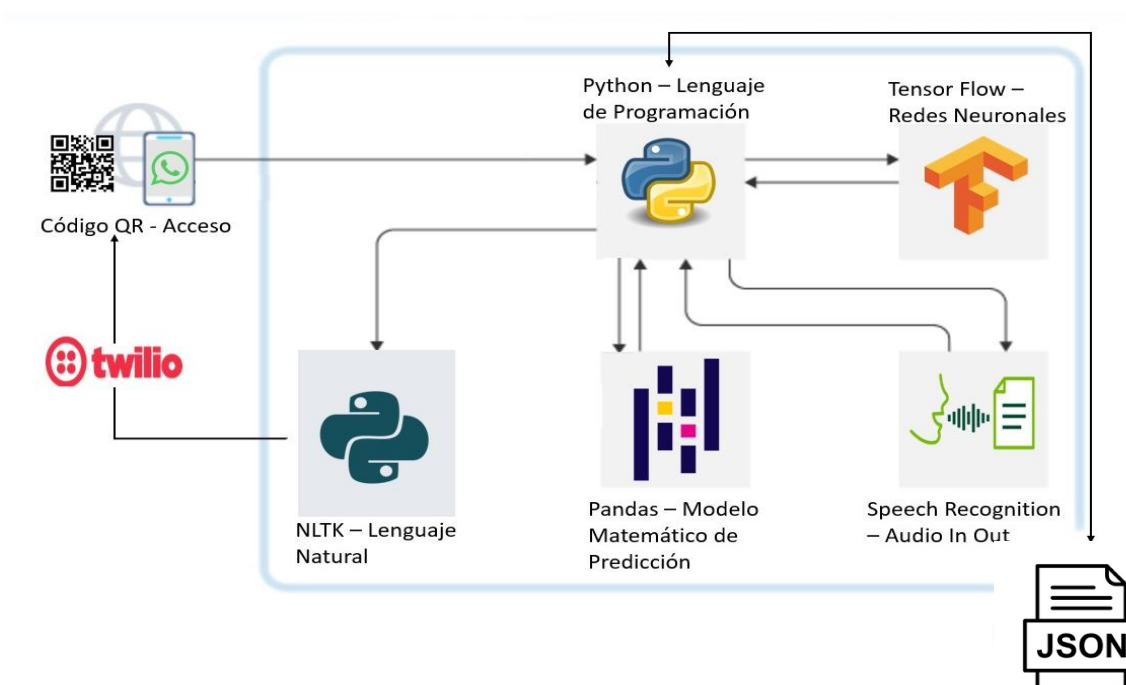
7.3.5 Arquitectura de alto nivel

En el siguiente diagrama se explica la arquitectura proporcionada en una descripción general del sistema, donde se identificaron los componentes principales para desarrollar el producto (chatbot)

El cliente tendrá acceso al chatbot a través de un código QR que leerá e inmediatamente iniciará una conversación por medio del servicio de mensajería Twilio quien entra una conexión directa al chatbot mediante el componente desarrollado con Python y establecerá una relación con la BD Pstgresql donde se encuentra información relevante del menú del establecimiento comercial.

Ilustración 9

Arquitectura de alto nivel



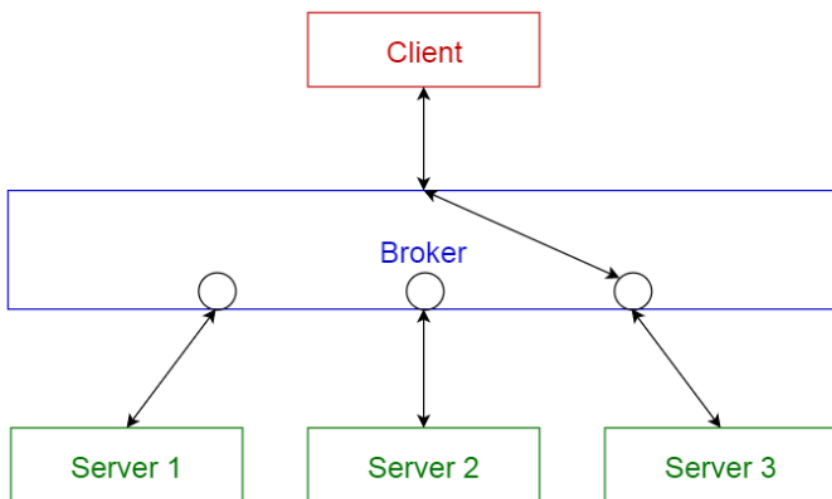
Fuente: Diseño propio

8. Construcción

En esta etapa del proyecto se ha evaluado que el patrón de arquitectura que se acopla mejor al diseño de este es el que está basado en la arquitectura de Broker, ya que se implementara un sistema de software distribuido con un componente desacoplado que debe interactuar con invocación de servicios remotos, en este caso usaremos el servicio de mensajería de Twilio.

Ilustración 10

Patrón arquitectura Broken



Fuente: <https://aselarbd.medium.com/message-brokers-and-brief-comparison-of-them-part-1-4c573cdcc50f>

Teniendo en cuenta los diagramas de Arquitectura de alto nivel que fueron construidos en la etapa de diseño se define que el proyecto se desarrollara con lenguaje Python que contiene las librerías necesarias para realizar en entrenamiento de la red neuronal como lo son el Tensor

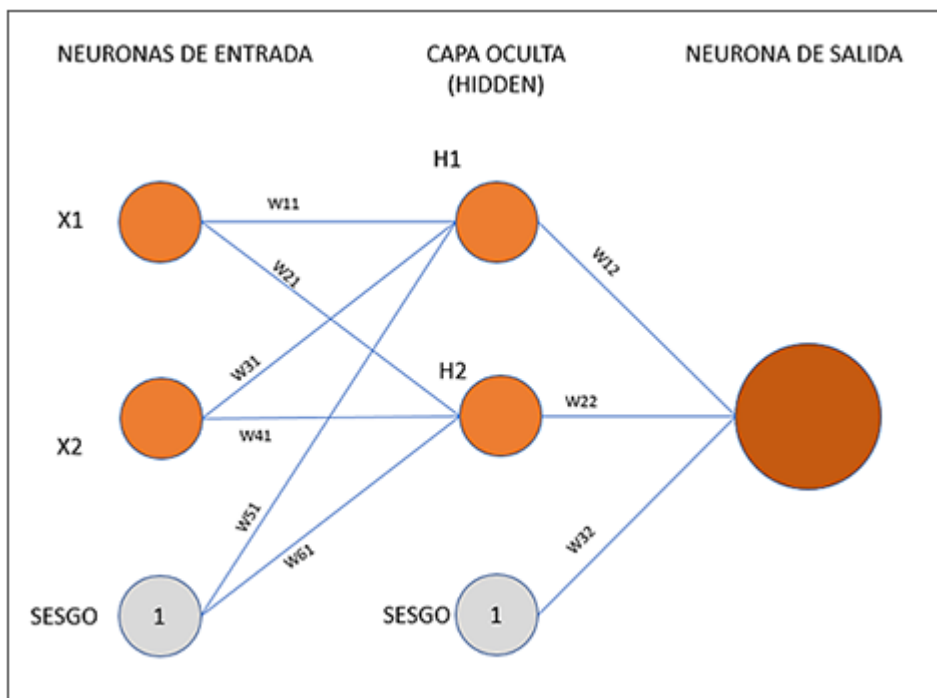
Flow, Speech Recognition, Pandas y el Lenguaje Natural NLTK, la utilización de estas librerías permitirá al proyecto realizar el entrenamiento de la red neuronal.

El entrenamiento de la red neuronal usando la librería de TensorFlow se basa esencialmente en una lista de entradas y un resultado, creando un modelo que toma las entradas y los resultados esperados de cada entrada y pueda aprender por sí solo el algoritmo necesario para hacer la conversión.

Estas redes neuronales se separan en capas, cada capa puede tener una o más neuronas, cualquier red siempre tiene al menos una capa de entrada que es donde se reciben los datos de entrada, y una capa de salida donde obtendremos el resultado calculado, también tenemos capas intermedias llamadas capas ocultas. Las neuronas se conectan a través de conexiones conectando las neuronas de entradas y las salidas, cada conexión tiene un peso asignado este peso es un valor numérico que representa la importancia de la conexión entre las neuronas. Cada neurona conexión de la capa de entrada tiene un sesgo que también un valor numérico y por último está el concepto de función de activación.

Ilustración 11

Estructura de una red neuronal



Fuente:

<https://www.ediciones-eni.com/open/mediabook.aspx?idR=8464c1bddd447da5e04304b958a18be>

Para poder entrenar la red lo primero que se necesita es tener muchos ejemplos de entradas y resultados, le damos a la red los ajustes automáticos de los pesos y sesgos, para que pueda hacer las predicciones lo más acertadas posibles, para lograrlo la red toma todos los datos y por cada uno hace la predicción, dependiendo de los resultados ajustara los pesos y sesgos.

Lo primero que necesitamos para la implementación de la red neuronal es la instalación de TensorFlow para el lenguaje de programación seleccionado en este caso el lenguaje Phyton, en la siguiente imagen vemos la instalación correcta de la librería:

Ilustración 12

Instalación librería TensorFlow

```

Microsoft Windows [Versión 10.0.19043.1586]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Admin>python --version
Python 3.7.4

C:\Users\Admin>pip install tensorflow==1.15
Requirement already satisfied: tensorflow==1.15 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (1.15.0)
Requirement already satisfied: wheel>=0.26 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (0.37.0)
Requirement already satisfied: termcolor>=1.1.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.1.0)
Requirement already satisfied: opt-einsum>=2.3.2 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (3.3.0)
Requirement already satisfied: tensorflow-estimator==1.15.1 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.15.1)
Requirement already satisfied: wrapt>=1.11.1 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.12.1)
Requirement already satisfied: keras-applications>=1.0.8 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.0.8)
Requirement already satisfied: tensorboard<1.16.0,>=1.15.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.15.0)
Requirement already satisfied: protobuf>=3.6.1 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (3.18.1)
Requirement already satisfied: astor>=0.6.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (0.8.1)
Requirement already satisfied: google-pasta>=0.1.6 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (0.2.0)
Requirement already satisfied: keras-preprocessing>=1.0.5 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.1.2)
Requirement already satisfied: numpy<2.0,>=1.16.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.19.5)
Requirement already satisfied: absl-py>=0.7.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (0.14.1)
Requirement already satisfied: six>=1.10.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.15.0)
Requirement already satisfied: gast>=0.2.2 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (0.2.2)
Requirement already satisfied: grpcio>=1.8.6 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.41.0)
Requirement already satisfied: h5py in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from keras-applications>=1.0.8->tensorflow==1.15) (3.1.0)
Requirement already satisfied: markdown>=2.6.8 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorboard<1.16.0,>=1.15.0->tensorflow==1.15) (3.3.4)
Requirement already satisfied: setuptools>=41.0.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorboard<1.16.0,>=1.15.0->tensorflow==1.15) (58.2.0)
Requirement already satisfied: werkzeug>=0.11.15 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorboard<1.16.0,>=1.15.0->tensorflow==1.15) (2.0.2)
Requirement already satisfied: cached-property; python_version < "3.8" in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from h5py->keras-applications>=1.0.8->tensorflow==1.15) (1.5.2)
Requirement already satisfied: importlib-metadata; python_version < "3.8" in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from markdown>=2.6.8->tensorboard<1.16.0,>=1.15.0->tensorflow==1.15) (4.8.1)
Requirement already satisfied: typing-extensions>=3.6.4; python_version < "3.8" in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from importlib-metadata; python_version < "3.8"->markdown>=2.6.8->tensorboard<1.16.0,>=1.15.0->tensorflow==1.15) (3.7.4.3)
Requirement already satisfied: zipp>=0.5 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from importlib-metadata; python_version < "3.8"->markdown>=2.6.8->tensorboard<1.16.0,>=1.15.0->tensorflow==1.15) (3.6.0)
You are using pip version 19.0.3, however version 22.0.4 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

```

Fuente: Diseño propio

Después de la instalación correcta de la librería, se genera el código necesario para programar la red neuronal.

Ilustración 13

Código fuente entrenamiento red neuronal

```

entrenamiento = numpy.array(entrenamiento)
salida = numpy.array(salida)

with open ("variables.Pickle", "wb") as archivoPickle:
    pickle.dump((palabras, tags, entrenamiento, salida), archivoPickle)

tensorflow.reset_default_graph()
red = tflearn.input_data(shape=[None, len(entrenamiento[0])])
red = tflearn.fully_connected(red, 100)
red = tflearn.fully_connected(red, len(salida[0]), activation="softmax")
red = tflearn.regression(red) #probabilidades
modelo = tflearn.DNN(red)
modelo.fit(entrenamiento, salida, n_epoch=1000, batch_size=100, show_metric=True)
modelo.save("modelo.tflearn")

entrada = str(msg)
cubeta = [0 for _ in range(len(palabras))]
entradaProcesada = nltk.word_tokenize(entrada)
entradaProcesada = [stemmer.stem(palabra.lower()) for palabra in entradaProcesada]
for palabraIndividual in entradaProcesada:
    for i, palabra in enumerate(palabras):
        if palabra == palabraIndividual:
            cubeta[i]=1
resultados =modelo.predict([numpy.array(cubeta)])
resultadosIndices = numpy.argmax(resultados)
global tag
tag = tags[resultadosIndices]
print(tag, "-----")
men.append(tag)
#Primera intencion del bot saludo

mens = len(men)
print(mens, "-----")
j = men[mens-1]
print(j, "////////////////////////////////////")

if "Saludo" == j:
    print ("holaaaaaa")

```

Fuente: Diseño propio.

Se establece un código QR para que el usuario tenga acceso al chatbot al ingresar al gastrobar, este se crea de manera personalizada utilizando código Python con la librería qrcode y pillow teniendo en cuenta aspectos importantes de marca como colores, tamaño y logo.

Ilustración 14

Construcción código QR

```
import qrcode
from PIL import Image

imagenlogo = r'C:\Users\Admin\Documents\robobar.PNG'
logo = Image.open(imagenlogo)

# tamaño de la imagen
hsize = int((float(logo.size[1])*float(100/float(logo.size[0]))))
logo = logo.resize((100, hsize), Image.ANTIALIAS)
QRcode = qrcode.QRCode(error_correction=qrcode.constants.ERROR_CORRECT_H)

# datos el código QR
url = 'wa.link/ap0rdz' #URL del Código
QRcode.add_data(url)
QRcode.make

# color al código y un color al fondo
QRcolor = 'Green'
QRfondo = 'White'

# nuestra imagen al código QR
QRimg = QRcode.make_image(fill_color=QRcolor, back_color=QRfondo).convert('RGB')

# posicion de la imagen, en este caso - centro.
pos = ((QRimg.size[0] - logo.size[0]) // 2, (QRimg.size[1] - logo.size[1]) // 2)
QRimg.paste(logo, pos)

# Guardamos la imagen de nuestro código QR en un directorio
QRimg.save(r'C:\Users\Admin\Documents\robobarqr.PNG')
```

Fuente: Diseño propio.

Es importante el diccionario de datos que para este proyecto se manejó en formato JSON. Esta parte es el corazón del proyecto ya que aquí es donde estamos entrenando la red neuronal a partir de palabras de reconocimiento clave y responder de manera muy acertada.

Básicamente el entrenamiento del chatbot se realiza de una manera muy parecida a como se le enseña a un niño, un ejemplo de esto es el saludo. Una persona puede saludar de múltiples maneras, sin embargo, al niño en un inicio no conoce ninguna por lo que hay que enseñarle esas maneras a partir de una palabra clave y así va reconociendo que lo que están generando es un saludo.

A continuación, está el diccionario de datos base de entrenamiento donde se garantiza un 100% de procesamiento de inteligencia artificial en los flujos del chatbot.

Ilustración 15

Construcción diccionario de datos

```
[{"contenido": [
  {"tag": "Saludo",
   "patrones": ["hola", "hello", "ola", "holi", "Buenas", "oli", "hi"]
  },
  {"tag": "Despedida",
   "patrones": ["adios", "chao", "nos vemos", "Adiós", "que le vaya bien", "gracias", "muchas gracias", "Adiós", "adiós"]
  },
  {"tag": "nada",
   "patrones": ["shfasjd", "kjhfaisjfpoad", "jsnjd", "lsjsdj", "iushdaoidaoidj"]
  },
  {"tag": "1",
   "patrones": ["1", "menu", "menú", "quiero ver el menu", "carta", "comida", "ver menu"]
  },
  {"tag": "2",
   "patrones": ["2", "mesero", "requiero al mesero", "ayuda", "atención", "atencion", "atension", "mezero", "help"]
  },
  {"tag": "3",
   "patrones": ["3", "cuenta", "pagar", "necesito la cuenta", "necesito pagar", "mi pago"]
  }
]
```

Fuente: Diseño propio.

Con la librería Pandas realizamos el manejo y análisis de estructuras de datos, es aquí donde leemos y escribimos fácilmente ficheros en formato CSV, Excel y bases de datos y JSON, accediendo a los datos mediante índices o nombres para filas y columnas, También se hace el reordenamiento, división y combinación de los conjuntos de datos, para trabajar con ellos en las respuestas del chatbot y para el diccionario de datos utilizado en el entrenamiento de la red neuronal. También con esta librería se realizan todas estas operaciones matemáticas que generan algún tipo de respuesta en el chatbot de una manera muy ordenada.

Con Speech recognition se realiza el tratamiento de audio tanto para solicitar en el chatbot una opción como para devolver una respuesta. Lo que se realiza aquí es una implementación donde a nivel de código se recibe un audio y este es transformado en un texto plano para ser tratado como

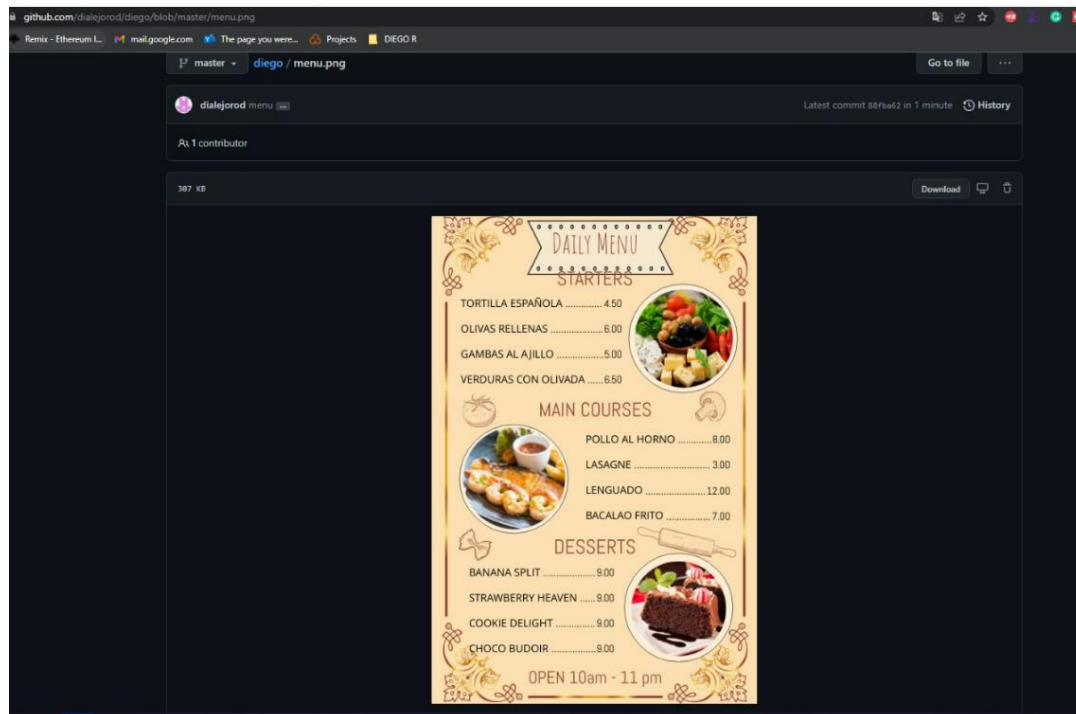
un mensaje ordinario en texto, luego este entra a la red neuronal para conocer el porcentaje de intención de la pregunta y posteriormente responderla.

Para poder humanizar el chatbot es importante conocer en qué momento se va a interactuar con contenido multimedia, sin embargo, para realizar este proceso es necesario tener en cuenta en donde se van a guardar las imágenes o contenido multimedia que se va a enviar, y como con código Python esta información se va a procesar y así mismo responderle al usuario final.

Utilizamos Github para guardar el archivo multimedia así:

Ilustración 16

Imágenes multimedia



Fuente: Diseño propio.

Luego lo procesamos con un fragmento de código Python así:

Ilustración 17

Código phyton

```
elif "2" == j:
    account_sid = 'AC9309637f363dd1379c8366d955013b29'
    auth_token = '04052c4aade2058c46ec7afb45d22f45'
    client = Client(account_sid, auth_token)
    from_whatsapp_number = 'whatsapp:+14155238886'
    to_whatsapp_number = num
    message = client.messages.create(body='',
                                     media_url='https://github.com/a4e9024b-b2f4-49d3-afe3-3b97a63',
                                     from_=from_whatsapp_number,
                                     to=to_whatsapp_number)
```

Fuente: Diseño propio.

Se ha definido que el manejo de datos se realizará con un modelo de procesamiento de datos que permite analizar sintácticamente, generar, transformar y procesar este tipo de datos desarrollado en lenguaje Python. Ya que el para proyecto no es importante el hecho de no disponer de procesamiento XSLT para manipular los datos en el cliente, ni es necesaria la captura de información del cliente Json nos ofrece practicidad a la hora de este intercambio de datos entre el cliente y el chatbot.

9. Pruebas

Las pruebas de software son parte integral del ciclo de vida del desarrollo, estas pruebas son la garantía que se presenta tanto para los stakeholder e interesados del proyecto de que el producto entregado cumple con las especificaciones del requerimiento.

De acuerdo con la metodología seleccionada para el proyecto las pruebas de calidad de software estarán a cargo del analista de pruebas y tendrá como objetivo principal asegurar la calidad de los componentes construidos, realizara pruebas funcionales sobre las historias de usuario y la ejecución de pruebas no funcionales que para este caso se realizarán enfocadas en pruebas de desempeño y seguridad.

Se han definido 3 fases diferentes de pruebas para el desarrollo del proyecto: pruebas de proceso, pruebas de producto y pruebas de seguridad.

9.1 Pruebas de Proceso

Para la realización de las pruebas del proceso se inició con una fase de pruebas de análisis estático de código utilizando SonarCloud como herramienta libre donde se evidenciaron los puntos susceptibles de mejora lo que permite la obtención de métricas necesarios para mejorar el código.

Ilustración 18

Medición código a través de SonarCloud

The dashboard shows the following metrics:

- Bugs:** 1 (C)
- Vulnerabilities:** 1 (C)
- Hotspots Reviewed:** 0.0% (E)
- Code Smells:** 21 (A)
- Duplications:** 0.0%
- Python:** 176 XS

Quality Gate: Not computed. The Quality Gate helps you see if your New Code is deployable or not.

RELIABILITY: 1 Bugs (C)

MAINTAINABILITY: 21 Code Smells (A)

SECURITY: 1 Vulnerabilities (C)

SECURITY REVIEW: 1 Security Hotspots (E), 0.0% Reviewed

COVERAGE: A few extra steps are needed for SonarCloud to analyze your code coverage. Setup coverage analysis.

DUPLICATIONS: 0.0% Duplications

Filters: Clear All Filters. 1/21 Issues. 1h 46min effort.

Type	CODE SMELL	Severity	Effort
Code Smell	Remove this commented out code. Why is this an issue?	Minor	23min effort
Code Smell	Rename function "getData" to match the regular expression "[a-z]_[a-z0-9]*". Why is this an issue?	Minor	18min effort
Code Smell	Refactor this function to reduce its Cognitive Complexity from 33 to the 15 allowed. Why is this an issue?	Critical	22min effort
Code Smell	Remove the unused local variable "responded". Why is this an issue?	Minor	5min effort
Code Smell	Remove the unused local variable "y". Why is this an issue?	Minor	5min effort
Code Smell	Specify an exception class to catch or rethrow the exception. Why is this an issue?	Critical	5min effort
Code Smell	Rename this local variable "auxX" to match the regular expression "[_a-z][a-z0-9]*". Why is this an issue?	Minor	2min effort
Code Smell	Rename this local variable "auxY" to match the regular expression "[_a-z][a-z0-9]*". Why is this an issue?	Minor	2min effort
Code Smell	Rename this local variable "auxPalabra" to match the regular expression "[_a-z][a-z0-9]*". Why is this an issue?	Minor	2min effort
Code Smell	Rename this local variable "salidaVec1a" to match the regular expression "[_a-z][a-z0-9]*". Why is this an issue?	Minor	2min effort
Code Smell	Rename this local variable "filaSalida" to match the regular expression "[_a-z][a-z0-9]*". Why is this an issue?	Minor	2min effort

Fuente: Diseño resultado análisis Sonar Cloud

Al estar en etapa de desarrollo se puede evidenciar que aún falta mejorar la calidad del código, ya que el resultado del análisis realizado arroja que hay 1 bugs, 21 code smells y un 0% de código duplicado.

Con el propósito de cumplir con los objetivos específicos del proyecto se ejecuta un proceso para cada uno de los mensajes recibidos por el ChatBot, en la siguiente grafica se evidencia que el tiempo de respuesta dado por el chatbot es de 1.24 segundos, se esperaba una demora de hasta 5 segundos en cada mensaje.

Ilustración 19

Tiempo de respuesta del chatBot

Message Segments

1

Encoding -

Cost -

Body

Hola Bienvenido a MI restaurante soy BOTTENDER y es un gusto poder ayudarte, a continuación tienes una serie de opciones que puedes elegir según lo que quieras consultar:

1. Selección del menú
 2. Solicitud de la cuenta
 3. Buzón de Sugerencias

Recuerda visitar nuestra página web <https://www.miRestaurante.co/>

Delivery Steps

STAGE	STEP	TIMESTAMP	ELAPSED TIME
Twilio Platform	Message Created	19:56:40 UTC 2022-04-28	—
Outbound Queue	Dequeued	19:56:40 UTC 2022-04-28	0.1 sec
	Enqueued	19:56:40 UTC 2022-04-28	0.08 sec
	Sent	19:56:41 UTC 2022-04-28	0.61 sec
Carrier Network	Message Read	19:56:41 UTC 2022-04-28	1.24 secs

Fuente: Resultado consulta consola Twilio.

9.2 Pruebas de Producto

Dentro de las pruebas de producto se estableció desarrollar las pruebas funcionales realizadas a los requerimientos funcionales para prevenir errores y validar los criterios de aceptación, estas pruebas fueron realizadas por el ProductOwner.

En la siguiente tabla se evidencia los criterios de aceptación aprobados por las partes:

Tabla 7*Crterios de aceptación*

ID	Criterio de Aceptación	Aceptado	Incidente	Corregido	Observación
1	Escaneo correcto de QR	X			
2	Redireccionamiento del código QR al WhastApp del cliente	X			
3	Inicio de conversación con el Chatbot	X			Al ser un ambiente de pruebas el inicio de la conversación empieza con un código que se envía a Twilio y esta contesta con el No. Al cual llegaran los mensajes.
4	Verificación de tiempo de respuesta de mensajes del ChatBot	X			Se evidencia la respuesta en menos de 1,2 segundos Ilustración No. 15
5	Recorrido por los diferentes menús	X			
5.1	Saludo	X			Ilustración No. 15
5.2	Saludo con error ortográfico	X			Ilustración No. 15
5.3	Selección de opción 1 "Selección del menú"	X			
5.3.1	Selección menú de comida o bebidas.		X	X	
5.3.2	Selección de pedido de comida	X			Incluida la imagen del plato
5.3.3	Selección de pedido de "Bebida"	X			Incluida la imagen de la bebida
5.4	Selección opción "Solicitud de cuenta"	X			
5.4.1	Selección opción de pago 1. Efectivo 2. Tarjeta		X	X	
5.5	Verificación de mensajes	X			

	automáticos de servicio				
5.6	Regreso al menú principal	X			
5.7	Iniciar una nueva conversación	X			

Fuente: Diseño Propio

A continuación, se mostrarán algunas imágenes del chatBot:

- Generación de código QR

Ilustración 20

Código QR

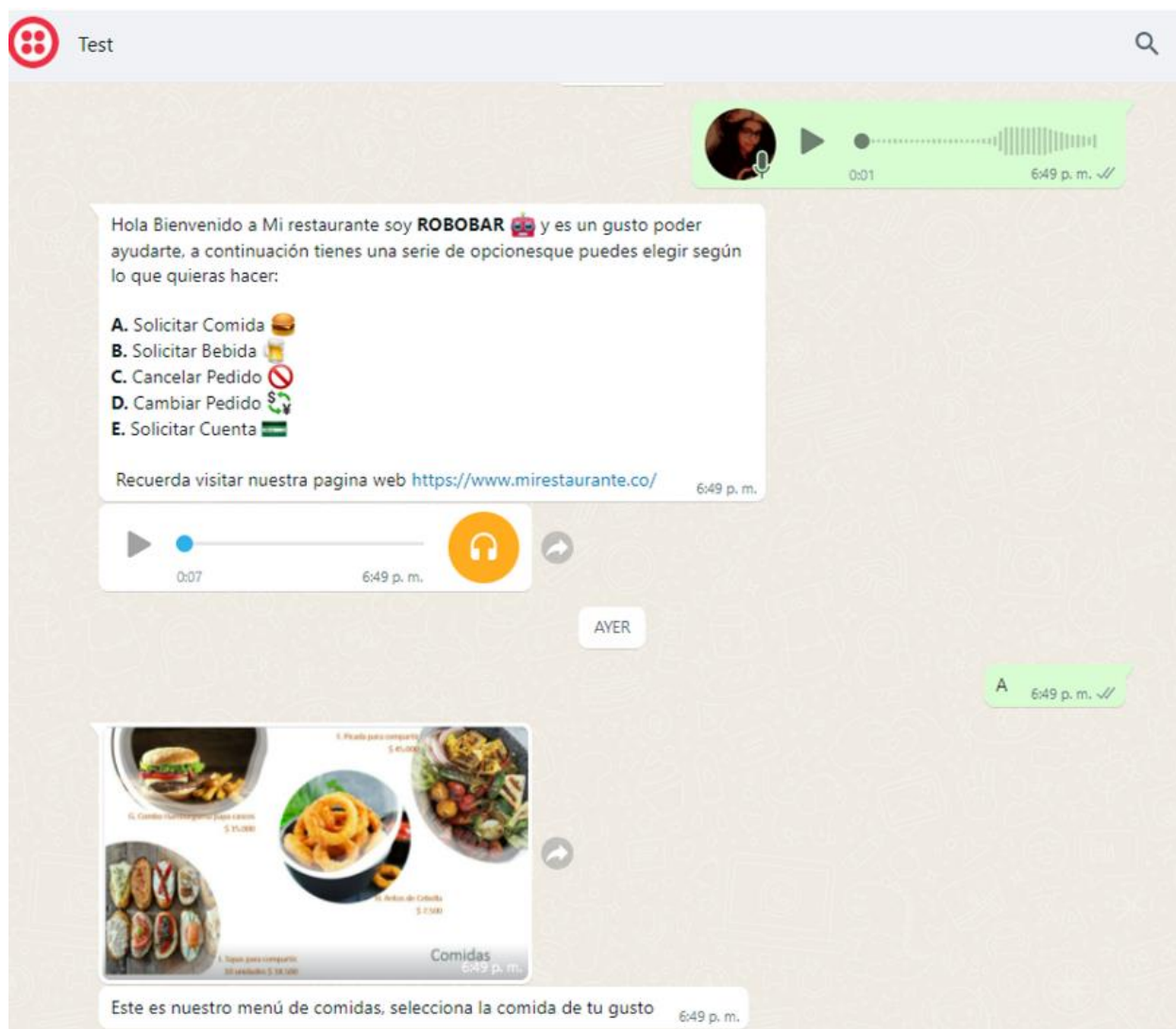


Fuente: Diseño propio

- Inicio de conversación con el ChatBot

Ilustración 21

Inicio conversacion en WhastApp



Fuente: Diseño propio

Ilustración 22

Imágenes flujo de conversación con el chatbot.

The image displays four sequential screenshots of a chatbot conversation. Each screenshot shows a chat interface with a header 'Test' and a search icon. The conversation progresses through several steps:

- First Screenshot:** The chatbot asks, "En letras o números indicame cuantas picadas quieres para tu pedido" (6:56 p. m.). The user replies "Picada" (6:56 p. m.). The chatbot then asks, "Indicame cuantas, seguido de la palabra pinta" (7:20 p. m.). The user replies "Dos" (6:56 p. m.). The chatbot confirms, "En 15 minutos tendrás tu **Picada** en la mesa 🍷🍷🍷" (6:56 p. m.).
- Second Screenshot:** The chatbot shows a menu titled "Bebidas" with various drink options and prices. The chatbot asks, "Este es nuestro menú de bebidas, selecciona la bebida de tu gusto" (7:17 p. m.). The user replies "B" (7:17 p. m.). The chatbot asks, "Indicame cuantas, seguido de la palabra pinta" (7:20 p. m.). The user replies "J" (7:20 p. m.). The chatbot asks, "Indicame cuantas, seguido de la palabra pinta" (7:20 p. m.). The user replies "Dos pintas" (7:20 p. m.). The chatbot confirms, "En 15 minutos tendrás tu Orden en la mesa." (7:20 p. m.).
- Third Screenshot:** The chatbot greets the user: "Hola Bienvenido a Mi restaurante soy **ROBOBAR** 🤖 y es un gusto poder ayudarte. a continuación tienes una serie de opciones que puedes elegir según lo que quieras hacer:" (7:25 p. m.). The chatbot lists options: "A. Solicitar Comida 🍽️", "B. Solicitar Bebida 🍷", "C. Cancelar Pedido 🚫", "D. Cambiar Pedido 🔄", "Z. Solicitar Cuenta 📄". The chatbot also provides a website link: "Recuerda visitar nuestra pagina web <https://www.mirestaurante.co/>" (7:25 p. m.). The user replies "Z" (7:25 p. m.). The chatbot provides a payment link: "En 5 minutos nuestro personal irá a tu mesa con las diferentes opciones de pago 💳💰 <https://pagos.mirestaurante.com/>" (7:25 p. m.).
- Fourth Screenshot:** The chatbot informs the user: "Tu decision me pone triste 😞 pero aun estas a tiempo de cancelar el pedido, en breve cancelaremos tu pedido" (7:08 p. m.). The user replies "C" (7:07 p. m.).

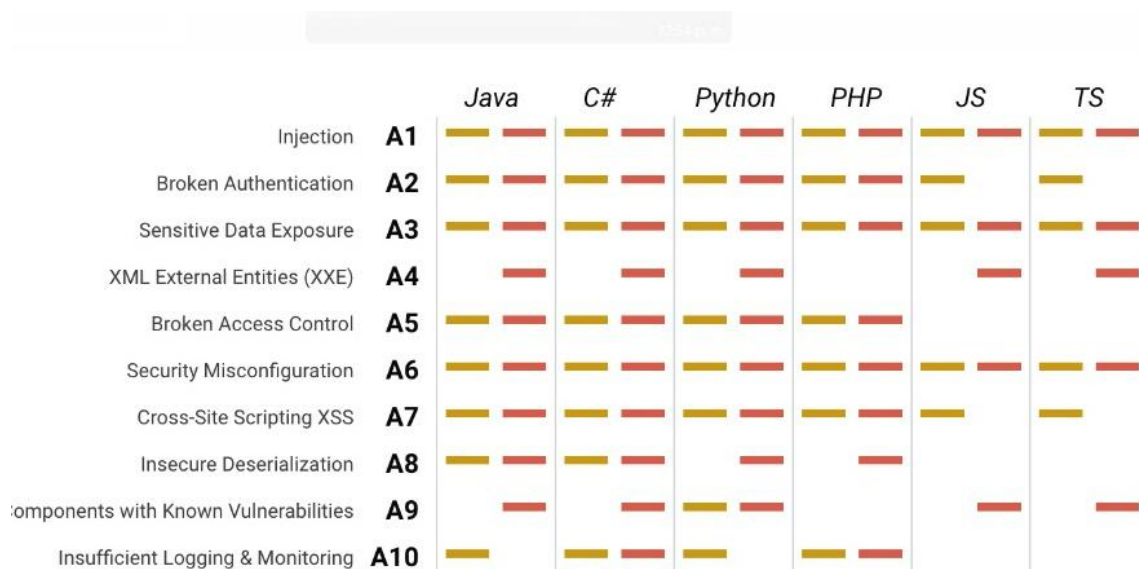
Fuente: Diseño propio

9.3 Pruebas de Seguridad

En esta etapa de pruebas de seguridad se utiliza la herramienta de SonarQube la cual permite analizar que la aplicación cumpla con el top 10 de OWASP. Lo cual se evidencia en la siguiente gráfica que el lenguaje de programación Python está incluida para su evolución.

Ilustración 23

Top 10 SonarQube



Fuente: Diseño Propio

Este chatbot como cualquier otro es susceptible al mismo tipo de vulnerabilidades, amenazas y riesgos que otras aplicaciones web orientadas al cliente. Las pruebas de seguridad realizada en este proyecto se enfocaron en la herramienta de SonarCloud y SonarQube donde se evidencio que al realizar en análisis el resultado arroja que a nivel de seguridad se cuenta con una incidencia y de funcionalidad nos da 0 de vulnerabilidades detectadas.

Ilustración 24

Análisis vulnerabilidades de funcionalidad y seguridad

Quality Gate ?

Not computed

The Quality Gate helps you see if your New Code is deployable or not.

Set New Code Definition

<p>RELIABILITY</p> <p>1 Bugs C</p>	<p>MAINTAINABILITY</p> <p>21 Code Smells A</p>
<p>SECURITY</p> <p>1 Vulnerabilities C</p>	<p>SECURITY REVIEW</p> <p>1 Security Hotspots 0.0% Reviewed E</p>
<p>COVERAGE</p> <p>A few extra steps are needed for SonarCloud to analyze your code coverage Setup coverage analysis</p>	<p>DUPLICATIONS</p> <p>0.0% Duplications C</p>

Filters Clear All Filters

▼ Type **VULNERABILITY** Clear

- Bug 1
- Vulnerability 1
- Code Smell 21

Bulk Change to select issues to navigate 1 / 1 Issues 30min effort

app.py

Change this code to not construct the URL from user-controlled data. Why is this an issue? 6 minutes ago · L70 · No tags

Vulnerability
Major
Open
dialejorod
30min effort
Comment

1 of 1 shown

Fuente: Diseño resultado análisis Sonar Cloud

10. Instalación y Configuración

Para esta fase del proyecto se realizara la explicación de los componentes necesarios y la correcta instalación de estos componentes y herramientas que se necesitan para el correcto funcionamiento del proyecto.

Para la realización del proyecto es necesario que la versión de Python sea 3.7 y de la librería Tensorflow es 1.15.

10.1 Instalación de las librerías Phyton

Para instalar estas librerías ya se debe tener instalado Python y Pip. Las librerías que se utilizaran para crear el Chat Bot son las siguientes: Requests, Tensorflow, Json, Tflern, Numpy, Flask, Twilio, Nltk, DateTime.

Se abre la consola y escribimos “pip install ‘el nombre de la librería. Como en el siguiente ejemplo:

Ilustración 25

Instalación librerías Phyton

```

Microsoft Windows [Versión 10.0.19043.1586]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Admin>python --version
Python 3.7.4

C:\Users\Admin>pip install tensorflow==1.15
Requirement already satisfied: tensorflow==1.15 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (1.15.0)
Requirement already satisfied: wheel<=0.26 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (0.37.0)
Requirement already satisfied: termcolor>=1.1.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.1.0)
Requirement already satisfied: opt-einsum>=2.3.2 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (3.3.0)
Requirement already satisfied: tensorflow-estimator<=1.15.1 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.15.1)
Requirement already satisfied: wrapt>=1.11.1 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.12.1)
Requirement already satisfied: keras-applications>=1.0.8 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.0.8)
Requirement already satisfied: tensorboard<1.16.0,>=1.15.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.15.0)
Requirement already satisfied: protobuf<=3.6.1 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (3.18.1)
Requirement already satisfied: astor>=0.6.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (0.8.1)
Requirement already satisfied: google-pasta>=0.1.6 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (0.2.0)
Requirement already satisfied: keras-preprocessing>=1.0.5 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.1.2)
Requirement already satisfied: numpy<2.0,>=1.16.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.19.5)
Requirement already satisfied: absl-py>=0.7.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (0.14.1)
Requirement already satisfied: six>=1.10.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.15.0)
Requirement already satisfied: gast>=0.2.2 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (0.2.2)
Requirement already satisfied: grpcio>=1.8.6 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorflow==1.15) (1.41.0)
Requirement already satisfied: h5py in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from keras-applications>=1.0.8->tensorflow==1.15) (3.1.0)
Requirement already satisfied: markdown>=2.6.8 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorboard<1.16.0,>=1.15.0->tensorflow==1.15) (3.3.4)
Requirement already satisfied: setuptools>=41.0.0 in c:\users\admin\appdata\local\programs\python\python37\lib\site-packages (from tensorboard<1.16.0,>=1.15.0->tensorflow==1.15) (58.2.0)

```

Fuente: Diseño propio

Observación: En algunos casos para instalar la librería de Json se tendrá que instalar de la siguiente manera “pip install simplejson”

10.2 Instalación de NGROK

Ngrok es una herramienta que permite acceder nuestro servidor local con una URL generada dinámicamente para acceder a internet, es utilizada por ejemplo cuando se necesita mostrar avances constantemente en sitios que se encuentran en etapa de desarrollo o cuando se trabaja con un equipo de desarrolladores de forma remota.

Se realiza la instalación descargando Ngrok desde el link <https://ngrok.com/download>

Ilustración 26

URL para descarga Ngrok

ngrok [CÓMO FUNCIONA](#) [PRECIOS](#) [DESCARGAR](#) [DOCS](#)
[SOLUCIONES EMPRESARIALES](#) [INICIAR SESIÓN](#) [REGISTRARSE →](#)

Descargar y configurar ngrok

Comience con ngrok en solo unos segundos.

- 1 Descargar ngrok**
 Primero, descargue el cliente ngrok, un binario único con cero dependencias de tiempo de ejecución.
[Download for Ventanas](#)
[Mac OS X](#) [Linux](#) [Mac \(32 bits\)](#)
[Windows \(32 bits\)](#) [Linux \(ARM\)](#)
[Linux \(ARM64\)](#) [Linux \(32 bits\)](#)
[FreeBSD \(64 bits\)](#) [FreeBSD \(32 bits\)](#)
- 2 Descomprimir para instalar**
 En Linux u OSX puede descomprimir ngrok desde una terminal con el siguiente comando. En Windows, simplemente haga doble clic en *ngrok.zip*.

```
$ unzip /path/to/ngrok.zip
```

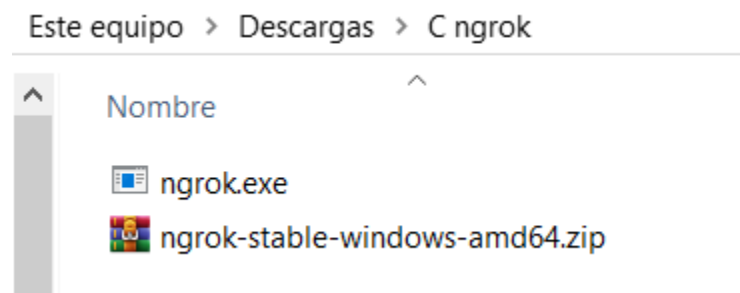

 A la mayoría de las personas les gusta mantener ngrok en su carpeta de usuario principal o establecer un alias para acceder fácilmente a la línea de comandos.

Fuente: <https://ngrok.com/download>

En descargas se encuentra el archivo comprimido que se de descarga en una carpeta para descomprimirlo.

Ilustración 27

Folder de descargas



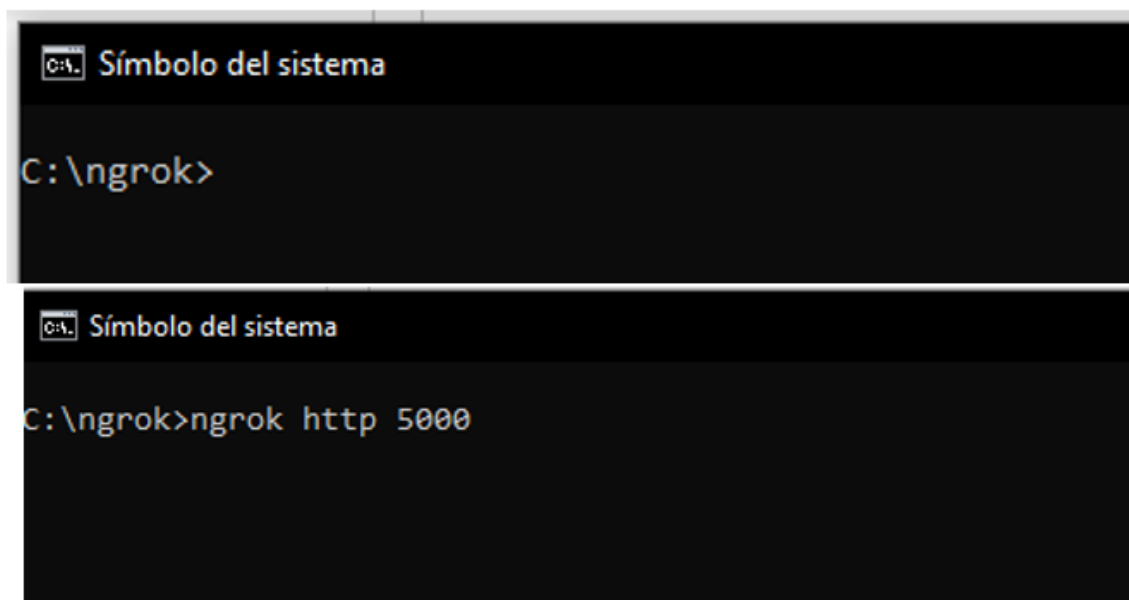
Fuente: Diseño propio

Desde CMD navegamos hasta la carpeta donde fue guardado Ngrok y se ejecuta el comando “ngrok http 5000”

Se debe tener en cuenta que se utiliza “5000” porque el equipo local nos da una respuesta como esta “ http://127.0.0.1:5000 ” así que se utilizara el número que esta después de “ : ” en este caso es “ 5000 ”

Ilustración 28

Ejecución Ngrok



The first screenshot shows a Windows command prompt window titled "Símbolo del sistema" with the current directory set to "C:\ngrok". The prompt is "C:\ngrok>".

The second screenshot shows the same window after the command "ngrok http 5000" has been entered, resulting in the prompt "C:\ngrok>ngrok http 5000".



The screenshot shows the output of the ngrok command in a terminal window titled "Símbolo del sistema - ngrok http 5000". The output is as follows:

```
ngrok by @inconshreveable
Session Status      online
Session Expires    7 hours, 59 minutes
Version            2.3.35
Region             United States (us)
Web Interface       http://127.0.0.1:4040
Forwarding          http://f900c1d363a1.ngrok.io -> http://localhost:5000
Forwarding          https://f900c1d363a1.ngrok.io -> http://localhost:5000

Connections        ttl    opn    rt1    rt5    p50    p90
                   0      0      0.00  0.00  0.00  0.00
```

Fuente: Diseño propio

Después de ejecutar las instrucciones se obtiene la URL para acceder a internet.

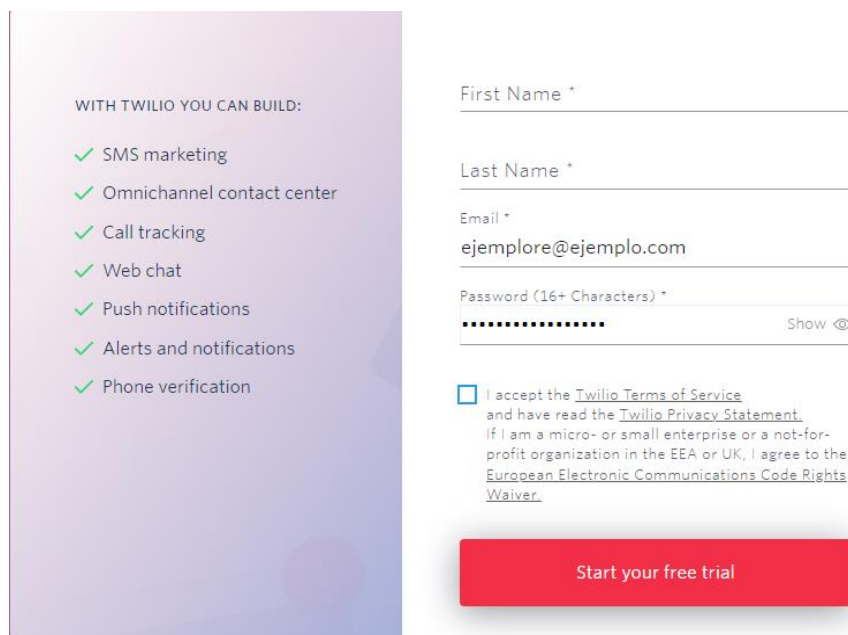
10.3 Conexión con Twilio

Twilio es una compañía que ofrece una plataforma de comunicaciones, así como de servicios en la nube, permite desarrollar aplicaciones que hagan y reciban llamadas, mensajes de texto, elaboren funciones de comunicación y registro, usando APIs, propias del servicio web.

Para la conexión con Twilio ingresamos al siguiente link: <https://www.twilio.com/es-mx/> donde es creada la cuenta:

Ilustración 29

Creación cuenta Twilio



WITH TWILIO YOU CAN BUILD:

- ✓ SMS marketing
- ✓ Omnichannel contact center
- ✓ Call tracking
- ✓ Web chat
- ✓ Push notifications
- ✓ Alerts and notifications
- ✓ Phone verification

First Name *

Last Name *

Email *
ejemplo@ejemplo.com

Password (16+ Characters) *
..... Show

I accept the [Twilio Terms of Service](#) and have read the [Twilio Privacy Statement](#). If I am a micro- or small enterprise or a not-for-profit organization in the EEA or UK, I agree to the [European Electronic Communications Code Rights Waiver](#).

Start your free trial

Fuente: Diseño propio

Una vez creada la cuenta se ingresa a la consola y se realizaran los ajustes necesarios a Messaging settings y a WhatsApp sandbox

Ilustración 30

Configuración cuenta Twilio

The image shows a screenshot of the Twilio console interface. At the top, the header includes 'Console', 'My first Twilio account', a trial status of '\$14.445 Upgrade', a search bar, and user information for 'Jorge'. The main content area is divided into two columns. The left column contains a sidebar with navigation options: 'Develop' and 'Monitor'. Under 'Develop', there are links for 'General', 'Log archives', 'Geo permissions', 'WhatsApp sandbox settings', 'Channels', and 'Add-ons* Beta'. Under 'Monitor', there are links for 'Voice' and 'Autopilot'. At the bottom of the sidebar is 'Docs and Support'. The right column displays the 'Account Info' section, which includes fields for 'Account SID' (AC8e04af0c10680927a22d3ee4b06cb18e), 'Auth Token' (masked with dots and a 'Show' button), and 'My Twilio phone number' (+17755227818). A warning icon and text state: 'Always store your token securely to protect your account. Learn more'. Below this, a note says: 'You are on a trial account. You can only send messages and make calls to [verified phone numbers](#). Learn more about your [trial account](#)'. To the right of the 'Account Info' section is a 'Helpful links' section with links for 'How does Twilio work?', 'API documentation', and 'Support help center'. At the bottom right of the console, there is a 'Refer a Friend' button.

▼ Settings

- General
- Log archives
- Geo permissions
- WhatsApp sandbox settings

Fuente: Diseño propio

En whatsapp sandbox se agrega la URL que se obtuvo de Ngrok.

Ilustración 31

Configuración Sandbox

The screenshot shows the Twilio console interface. At the top, there's a dark blue header with 'Console', 'My first Twilio account', a trial status of '\$14.445', and user information for 'Jorge'. A left sidebar contains navigation options like 'Studio', 'Functions', 'Phone Numbers', and 'Messaging'. The main area is titled 'Twilio Sandbox for WhatsApp' and 'Sandbox Configuration'. It contains two rows of configuration: 'WHEN A MESSAGE COMES IN' with a text input containing 'https://0f6c-186-86-33-61.r' and a dropdown set to 'HTTP Post'; and 'STATUS CALLBACK URL' with an empty text input and a dropdown set to 'HTTP Post'. Below this is the 'Sandbox Participants' section, which includes an instruction to invite friends and a table with one column labeled 'USERID' and the text 'No rows to display'.

Sandbox Configuration

To send and receive messages from the Sandbox to your Application, configure your endpoint URLs. [Learn more](#)

WHEN A MESSAGE COMES IN	<input type="text" value="https://0f6c-186-86-33-61.r"/>	HTTP Post
STATUS CALLBACK URL	<input type="text"/>	HTTP Post

Fuente: Diseño propio

Se agrega la URL ejemplo <https://0f6c-186-86-33-61.ngrok.io/sms>, adicionando al final la “sms”.

A partir de este momento ya se tiene la conexión y comunicación con el Chatbot a través de Twilio.

Ilustración 32

Conversación con ChatBot

The screenshot shows a chat interface with a header bar containing a red circular icon with a white grid, the text 'Test', and a search icon. The chat history includes:

- An audio message from the user, 0:01 in duration, timestamped 6:49 p. m. with a checkmark.
- A text message from the chatbot: "Hola Bienvenido a Mi restaurante soy **ROBOBAR** 🤖 y es un gusto poder ayudarte, a continuación tienes una serie de opciones que puedes elegir según lo que quieras hacer:"
- A list of five options:
 - A. Solicitar Comida 🍔
 - B. Solicitar Bebida 🍹
 - C. Cancelar Pedido 🚫
 - D. Cambiar Pedido 🔄
 - E. Solicitar Cuenta 📄
- A text message from the chatbot: "Recuerda visitar nuestra pagina web <https://www.mirestaurante.co/>" timestamped 6:49 p. m.
- An audio message from the chatbot, 0:07 in duration, timestamped 6:49 p. m., with a play button and a share icon.
- A separator message: "AYER"
- A text message from the user: "A" timestamped 6:49 p. m. with a checkmark.
- A menu image showing various food items with prices:
 - 1. Plato para compartir \$ 45,000
 - 2. Caudin (Caudin con queso y cebolla) \$ 17,000
 - 3. Arroz de Calabaza \$ 7,500
 - 4. Plato para compartir 30 unidades \$ 18,500
 - Comidas 6:49 p. m.
- A text message from the chatbot: "Este es nuestro menú de comidas, selecciona la comida de tu gusto" timestamped 6:49 p. m.

Fuente: Diseño propio

11. Conclusiones

El objetivo fundamental de este proyecto era crear una solución práctica y moderna para la gestión de los pedidos y el valor de la cuenta de los clientes en establecimientos comerciales como gastrobares utilizando aplicaciones de mensajería con respuestas automáticas implementando en ellas Inteligencia Artificial, para ofrecer una atención inmediata y prestar un servicio eficaz y eficiente. Este objetivo se pudo cumplir al 100% gracias a la implementación de las librerías que ofrece el lenguaje de programación Python y la utilización del diccionario de datos para poder realizar el entrenamiento de redes neuronales necesarias para emular una conversación a través del chatbot.

Se logró elaborar un chatbot que entregara respuestas inmediatas con una demora de aproximadamente 5 segundos, así mismo se consiguió disminuir en un 80% la espera que los comensales debían aguardar para realizar la solicitud del pedido o de la cuenta debido a que inmediatamente después del escaneo del código QR se generaba la conversación con el chatbot y el comensal podía realizar su pedido inmediatamente, esto se alcanzó gracias a que en la fabricación del código se utilizó un 60% de inteligencia artificial y el entrenamiento de la red neuronal.

Se entregó una solución que permite una interacción entre el chatbot y el comensal más humana, amena y divertida debido a la utilización de un 70% de los flujos con componentes multimedia, así como la entrega de mensajes automáticos a los clientes con información del gastrobar.

La implementación de esta solución resulta ser económica y sencilla tanto para los establecimientos comerciales como gastrobares y clientes ya que está pensada para usar una de

las herramientas de comunicación más utilizada a nivel mundial como es el WhatsApp, además de no ser necesario la descarga de ninguna aplicación adicional para poder usar el chatbot.

12. Anexo

12.1 Análisis de mercado y estudio de necesidades.

12.2 Implementación de políticas de seguridad.

12.3 Análisis de activos, riesgos, controles y efectividad.

12.4 Documento arquitectura SAD.

12.5 Documento de gestión de servicios para el proyecto.

13. Referencias

Aunoa. (2020) Arquitectura de un sistema de IA conversacional: Los 5 bloques esenciales.

Inteligencia Artificial. <https://aunoa.ai/arquitectura-de-un-sistema-de-ia-conversacional/>

Dasanayaka Asela. (Feb 16,2018). Message brokers and brief comparison of them - Part 1.

<https://aselarbd.medium.com/message-brokers-and-brief-comparison-of-them-part-1-4c573cdcc50f>

Drumond Claire. (2021). ¿Qué es ágil?. ATlassian Agile Coach.

<https://www.atlassian.com/es/agile>

Leao Luis. (2021). Crear un chatbot de WhatsApp con Python, Flask y Twilio.

<https://www.twilio.com/blog/crear-un-chatbot-de-whatsapp-con-python-flask-y-twilio>

Muradas Yanina. (2020). ¿Qué es Scrumban?. Metodologías y herramientas.

<https://openwebinars.net/blog/que-es-scrumban/>

2021. ¿Qué es un chatbot?. Oracle Digital Assistant. [https://www.oracle.com/co/chatbots/what-](https://www.oracle.com/co/chatbots/what-is-a-chatbot/)

[is-a-chatbot/](https://www.oracle.com/co/chatbots/what-is-a-chatbot/)

VANNIEUWENHUYZE Aurélien. (2020). Inteligencia Artificial Fácil - Machine Learning y

Deep Learning Prácticos. Eni.

Anexos

12.1 Análisis de mercado y estudio de necesidades

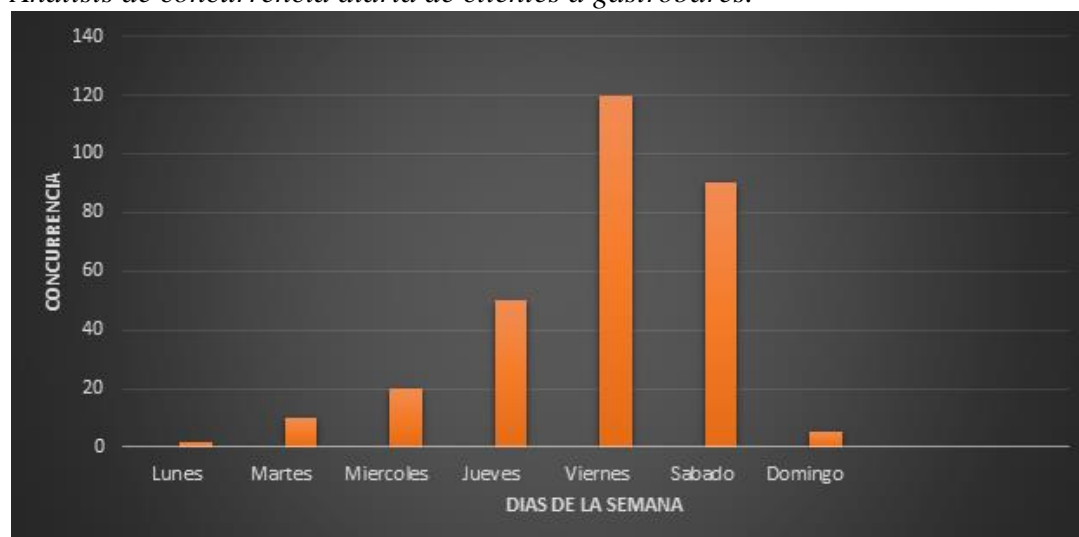
Análisis de negocio (PBA).

Para identificar los PBA del proyecto, visualizamos las actividades de negocio de los clientes en términos de su demanda de servicios. Se observan atributos como frecuencia, volumen, localización y duración.

De acuerdo con el análisis ejecutado se idéntica que el ciclo de mayor demanda de nuestro servicio son los fines de semana comenzando desde el jueves en horas de la noche con gran afluencia los viernes y sábados, ya que el son los días en la que la mayoría de las personas salen en plan de esparcimiento y entretenimiento.

Ilustración 1

Análisis de concurrencia diaria de clientes a gastrobares.



Fuente: Diseño Propio

Esta demanda va incrementando exponencialmente en la medida en que se implemente el proyecto en otros establecimientos comerciales en la ciudad de Bogotá, así mismo si el proyecto se amplía a ciudades que depende del turismo la tendencia indicaría que los fines de semana con festivos puede subir en estas ciudades turísticas a las visitas a los Gastrobares incluidos los días domingo.

Estudio de necesidades con método de observación.

Se realizaron varias visitas a diferentes establecimientos comerciales (Gastrobares) en los días más frecuentados identificados en el análisis de negocio (PBA) jueves, viernes y sábados en diferentes horarios, especialmente después de las 4 pm. Este estudio nos dio como resultado un promedio de demora en la atención a los clientes para realizar el pedido de aproximadamente un minuto 30 segundos.

También se evidencio que entre más clientes había en una sola mesa la solicitud al 100% del pedido se demoraba más en ser realizada.

Ilustración 2

Tablas de evaluación

P = Numero de personas
S = Segundos

Establecimiento Comercial No. 1				
	Mesa No.1	Mesa No. 2	Mesa No. 3	Mesa No. 4
Jueves	3	4	2	4
	60	60	50	40
Viernes	4	6	4	5
	50	120	60	100
Sabado	6	6	5	4
	120	120	80	100
Total Mesas * Dia	12			
Total Segundos	960			
Promedio	80,00			

Establecimiento Comercial No. 2				
	Mesa No.1	Mesa No. 2	Mesa No. 3	Mesa No. 4
Jueves	2	3	4	4
	60	60	70	80
Viernes	5	6	4	5
	60	120	60	100
Sabado	6	6	5	4
	120	120	80	100
Total Mesas * Dia	12			
Total Segundos	1030			
Promedio	85,83			

Establecimiento Comercial No. 3				
	Mesa No.1	Mesa No. 2	Mesa No. 3	Mesa No. 4
Jueves	4	6	6	4
	120	80	100	80
Viernes	2	6	4	5
	80	120	60	100
Sabado	6	6	5	4
	120	120	80	100
Total Mesas * Dia	12			
Total Segundos	1160			
Promedio	96,67			

Fuente: Diseños propios

Al generar la suma de los segundos de cada uno de los establecimientos y mesas se obtiene un resultado de 262.5 segundos y generar la orden de pedido al 100% para cada cliente para un promedio total de atención de aproximadamente **87.5 segundos por mesa**. Con el chatbot se pretende disminuir ese tiempo a mínimo 5 segundos por mesa.

12.2 Implementación de políticas de seguridad

Se anexa la relación de la implementación de las políticas de seguridad utilizando los siguientes dominios y controles del estándar ISO 27002:

Numero	Dominios	Control	Implementación
A.5	Políticas de seguridad de la información		
A.5.1	Dirección de gestión para la seguridad de la información		
A.5.1.1	Políticas para la seguridad de la información.	Definir un conjunto de políticas para la seguridad de la información, aprobada por la dirección, publicada y comunicada a los empleados y partes externas pertinentes.	<ol style="list-style-type: none"> 1. Elaborar política de control de accesos a la información relevante del negocio. 2. Elaborar una política clara de la transferencia de la información. 3. Establecer una política de privacidad y protección de la información personal identificable. 4. Publicar dichas políticas visible para empleados y clientes del establecimiento.
A.5.1.2	Revisión de las políticas para seguridad de la información	Las políticas para seguridad de la información se deberán revisar a intervalos planificados o si ocurren cambios significativos, para asegurar su idoneidad, adecuación y eficacia continuas.	<ol style="list-style-type: none"> 1. Desarrollar una política con distintos documentos con una estructura jerárquica. 2. Elaborar los manuales respectivos y capacitar al personal en el manejo de la información.
A.6	Organización de la información		
A.6.1	Organización interna		
A.6.1.1	Roles y responsabilidades para la seguridad de la información	Todas las responsabilidades de seguridad de la información deben definirse y asignarse.	<ol style="list-style-type: none"> 1. Elaborar un organigrama identificando cada rol y su responsabilidad, incluyendo los clientes y proveedores.

			<ol style="list-style-type: none"> 2. Publicar dicho organigrama. 3. Generar doble factor de seguridad en los repositorios donde se guarda la información.
A.6.2	Dispositivos móviles y teletrabajo		
A.6.2.1	Políticas para dispositivos móviles	Adaptar una política y medidas de seguridad de apoyo para gestionar los riesgos introducidos por el uso de dispositivos móviles.	<ol style="list-style-type: none"> 1. El establecimiento debe contar con mínimo dos dispositivos móviles registrados a nombre del establecimiento y asignados por roles administrador y chef. 2. Registro de software instalado en los dispositivos móviles. 3. Control de acceso a este dispositivo con registro de password para cada dispositivo. 4. Utilizar herramientas para bloqueo aplicaciones y páginas que no hagan parte de las actividades del proyecto.
A.7	Seguridad en Recursos humanos		
A.7.2	Durante la ejecución del empleo		
A.7.2.1	Responsabilidades de la dirección	La dirección debería exigir a todos los empleados y contratistas la aplicación de la seguridad de la información de acuerdo con las políticas y procedimientos establecidos por la organización.	<ol style="list-style-type: none"> 1. Mensualmente se impartirá una capacitación en seguridad de la información por parte de la dirección y deberá ser registrada en un formato definido para tal acción y firmada por cada empleado.
A.7.2.2	Toma de conciencia, educación y formación en la seguridad de la información	Todos los empleados de la organización, y en donde sea pertinente, los contratistas, deberían recibir la educación y la	<ol style="list-style-type: none"> 1. Se gestiona un modelo de certificación con los diferentes partners para que los empleados del establecimiento

		formación en toma de conciencia apropiada, y actualizaciones regulares sobre las políticas y procedimientos pertinentes para su cargo.	identifiquen los riesgos de seguridad y su mitiguen.
A.7.2.3	Proceso disciplinario	Se debería contar con un proceso disciplinario formal el cual debería ser comunicado, para emprender acciones contra empleados que hayan cometido una violación a la seguridad de la información.	1. Talento humano difundirá esta política y la hará cumplir de acuerdo a las acciones legales.
A.7.3	Terminación o cambio de empleo		
A.7.3.1	Terminación o cambio de responsabilidades de empleo	Las responsabilidades y los deberes de seguridad de la información que permanecen validos después de la terminación o cambio de contrato se deberían definir, comunicar al empleado o contratista y se deberían hacer cumplir.	1. Incluir como requisitos de desvinculación las responsabilidades legales e incluidas en los acuerdos de confidencialidad. 2. Generar un proceso automático que elimine las credenciales de las personas que estén involucradas en el proyecto y se retiren por diferentes motivos.
A.9	Control de Accesos		
A.9.4	Control de accesos a sistemas y aplicaciones		
A.9.4.1	Restricción de acceso a la información	El acceso a la información y a las funciones de los sistemas de las aplicaciones se debería restringir de acuerdo con la política de control de acceso.	1. Establecer las restricciones correspondientes de forma selectiva de derechos de lectura / escritura / eliminación y data relevante para el negocio. 2. Implementar herramientas que generen la restricción de acceso según el rol de cada empleado.

A.9.4.2	Procedimiento de ingreso seguro	Cuando lo requiere la política de control de acceso, el acceso a sistemas y aplicaciones se debería controlar mediante un proceso de ingreso seguro.	<ol style="list-style-type: none"> 1. Establecer límites de horas para el acceso a las aplicaciones. 2. Establecer el uso de VPN para el acceso a las aplicaciones y data del proyecto.
A.9.4.3	Sistema de gestión de contraseñas	Los sistemas de gestión de contraseñas deben ser interactivos y garantizar la calidad de las contraseñas.	<ol style="list-style-type: none"> 1. Hacer cumplir que cada rol tenga su contraseña definida para mantener la responsabilidad. 2. Establecer cambios de contraseñas en forma periódica.
A.9.4.4	Uso de programas utilitarios privilegiados	Restringir y controlar estrictamente el uso de programas utilitarios que pudieran tener capacidad de anular el sistema y los controles de las aplicaciones.	<ol style="list-style-type: none"> 1. Limitar la instalación o el acceso a programas diferentes a los necesarios en los dispositivos móviles. 2. Establecer los roles de administrador de los equipos únicamente al gerente del proyecto.
A.9.4.5	Control de acceso a códigos fuente de programas	Restringir el acceso a los códigos fuente de los programas.	<ol style="list-style-type: none"> 1. El personal de soporte no debe tener acceso a las librerías fuentes del código. 2. Definir una herramienta de gestión de repositorios segura y con doble factor de autenticación. 3. Usar herramientas que detecten información que pueda llegar a ser vulnerable y de ser así esta información deberá ser encriptada.

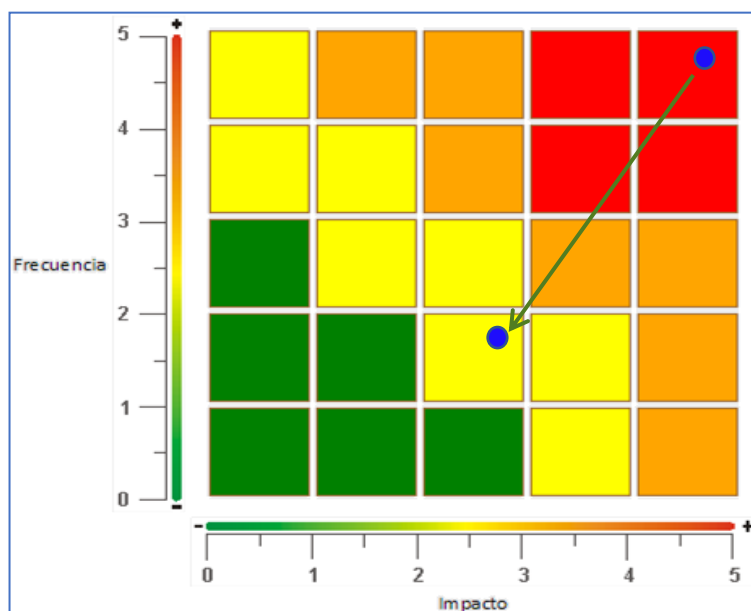
12.3 Análisis de activos, riesgos, controles y efectividad

El propósito de este documento es realizar un análisis de riesgos de los diferentes activos de este proyecto y ejecutar un ejercicio de cuales controles se deben implementar entonces obtener como resultado el nivel de efectividad de dichos controles a través de los mapas de calor.

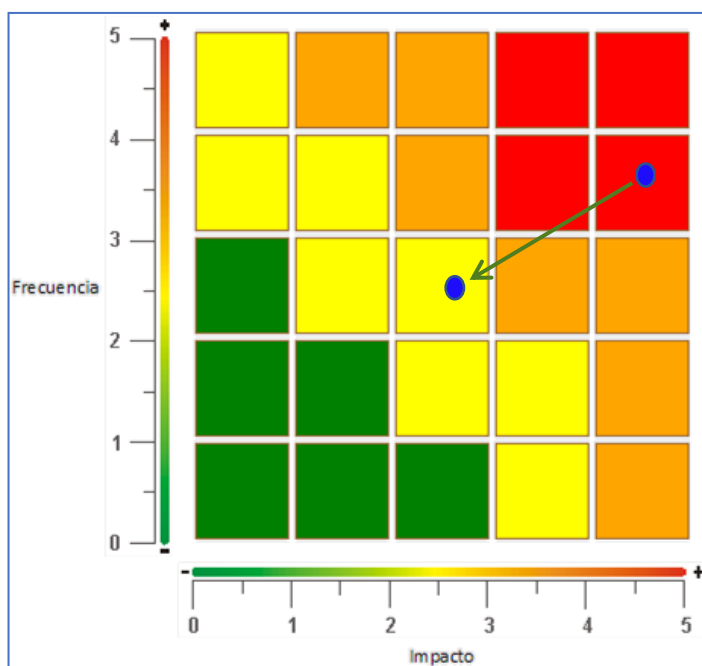
Nombre del activo	Descripción	Tipo de Activo	Clase de Activo	Propietario	Custodio	Confidencialidad	Disponibilidad		Integridad		Confidencialidad		IMP	FRE	RIE	Nivel de Riesgo
							Errores y Alteraciones	Frecuencia	Errores y Alteraciones	Frecuencia	Errores y Alteraciones	Frecuencia				
Base de datos	Contiene la información relevante para el manejo de la data necesaria para el proyecto.	Información	Base de datos	Dueño del restaurante	TI	Confidencial	5	4	5	3	5	3	5,00	3,33	16,67	Alto
Repositorio de código fuente	Contiene el código funcional del proyecto.	Información	Servidor	DEV	TI	Confidencial	5	4	5	3	4	4	4,67	3,67	17,11	Alto
Plataforma de mensajería	Contiene la información correspondiente a las respuestas generadas por la interacción del chatbot.	Información	Aplicación	Gerente del proyect	TI	Interna	5	3	4	2	3	2	4,00	2,33	9,33	Bajo
Documentos de proyecto	Contiene información correspondiente al levantamiento de requerimientos, actas	Información	Documentación	Gerente del proyecto	Gerente del proyecto	Interna	2	1	2	3	2	3	2,00	2,33	4,67	Bajo
Servidor de APP	Servidor donde estara alojada la app en producción	Hardware	Servidor	TI	TI	Interna	5	3	4	5	3	5	4,00	4,33	17,33	Alto
Aplicación de mensajería instantanea	Contiene la información de los clientes que se relacionaran con el proyecto	Software	Aplicación	Whastapp	Empresa de mensajería	Confidencial	5	5	5	4	5	4	5,00	4,33	21,67	Extremo
Jira software	Herramienta en línea para la administración de tareas del proyecto	Software	Sitios web	Atlassian	TI	Interna	4	3	3	2	4	3	3,67	2,67	9,78	Bajo
Menú restaurante	Contiene toda la data relacionada al menú ofrecido por el gastrobar.	Información	Carpeta compartida	Dueño del restaurante	TI	Interna	2	2	4	4	4	4	3,33	3,33	11,11	Moderado
Correo electronico	Contiene el correo electronico asociado a la cuenta de Twilio	Software	Aplicación	Todos los procesos	TI	Confidencial	5	2	3	2	3	2	3,67	2,00	7,33	Bajo
Acuerdos de confidencialidad	Contiene todos los documentos relacionados y firmados con los acuerdos de confidencialidad de parte de Gerencia de proyectos e información relevante del negocio.	Información	Documentación	Dueño del restaurante	Gerente del proyecto	Confidencial	2	1	2	3	2	3	2,00	2,33	4,67	Bajo

Riesgo	Activo de Información	Causas	Consecuencias	Frecuencia	Impacto	Riesgo Inherente	No.	Nombre de Control	Descripción del Control	Clase	Tipo	Valoración por Clase y Tipo	Valoración por Control	Efectividad de Control
Envío de información fraudulenta, spam, malware, etc a los clientes del establecimiento comercial.	Aplicación de mensajería instantánea	<ul style="list-style-type: none"> * Acceso no autorizado a bases de datos. * Acceso no autorizado a código fuente y ejecutables del chatbot. * Captura de información personal del cliente como el número de su Whastapp. 	<ul style="list-style-type: none"> * Quejas de clientes. * Pérdida de servicio en el establecimiento. 	4,33	5	Extremo	1	Gestión de políticas de roles para acceso a DB y Código fuente	La gerencia del proyecto identificará los roles y la asignación de estos al personal del proyecto.	Preventivo	Manual	4	15	Excelente
							2	Políticas de password	Director de desarrollo generará la automatización de cambio de passwords para los diferentes roles	Preventivo	Automático	6		
							3	Gestión de riesgos introducidos por el uso de dispositivos móviles	La gerencia de proyecto registrará a través de herramientas y app el software instalado en los dispositivos móviles.	Preventivo	Semiautomático	5		
Fuga y/o pérdida de código fuente relacionado con el proyecto.	Repositorio de código fuente	<ul style="list-style-type: none"> * No tener control sobre los usuarios y sus diferentes roles dentro del proyecto. * Error humano 	<ul style="list-style-type: none"> * Pérdida del proyecto. * Retrabajo. 	3,67	4,6	Alto	1	Gestión de control de acceso, el acceso a sistemas y aplicaciones se debería controlar mediante un proceso de ingreso seguro.	El director de IT generará doble factor de seguridad en los repositorios donde se guarda la información.	Preventivo	Manual	4	11	Bueno
							2	Restringir el acceso a los códigos fuente de los programas.	El director de IT establecerá los límites de horarios para el acceso a las aplicaciones usando herramientas para esta tarea.	Preventivo	Manual	4		
							3	Gestión de copias de seguridad de información	El director de IT establecerá herramientas para generar copias de seguridad, backups relacionados al código fuente.	Correctivo	Semiautomático	3		
Generar una indisponibilidad del servicio y fuga de información que exista en los logs de las app.	Servidor de APP	<ul style="list-style-type: none"> * Ataque informático al servidor. * Error humano. 	<ul style="list-style-type: none"> * Finalización del contrato por pérdida del servicio. 	4,33	4	Alto	1	Gestión de control de acceso a repositorios.	El director de IT establecerá el uso de VPN para el acceso a las aplicaciones y data del proyecto.	Preventivo	Semiautomático	5	16	Excelente
							2	Balanceador de servicio	El director de IT establecerá protocolos de balanceadores de servicio utilizando las herramientas que estén disponibles.	Preventivo	Automático	6		
							3	Herramientas de monitoreo de Logs	El director de IT establecerá la instalación y configuración de herramientas para análisis de log y control de las vulnerabilidades que puedan existir en los logs. El desarrollo se realizará de forma segura encriptando usuarios y contraseñas que existan en los logs.	Preventivo	Semiautomático	5		

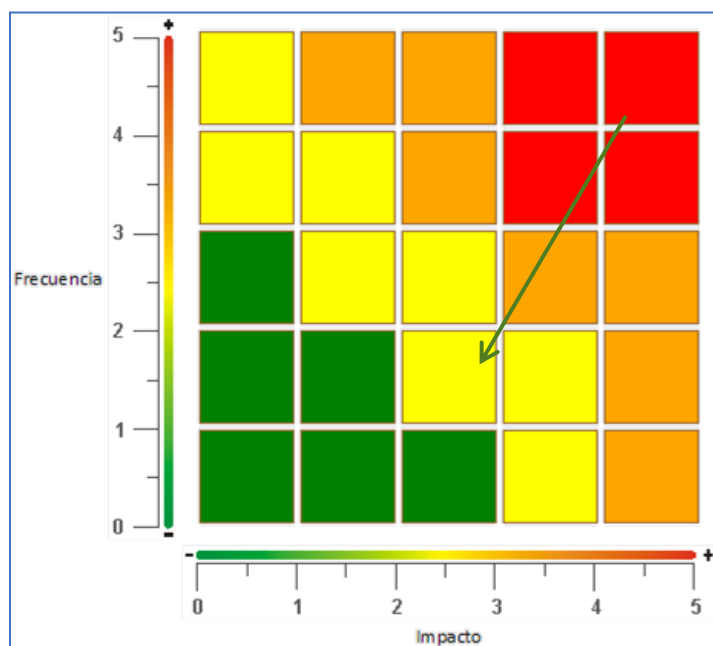
Riesgo No. 1 Envío de información fraudulenta, spam, malware, etc a los clientes del establecimiento comercial.



Riesgo No. 2 Fuga y/o perdida de código fuente relacionado con el proyecto.



Riesgo No. 3 Generar una indisponibilidad del servicio y fuga de información que exista en los logs de las app.



12.4 Documento arquitectura SAD

Introducción

El presente documento permite identificar aspectos y contenidos del documento de arquitectura de software, las especificaciones planteadas en el presente archivo están orientadas hacia el desarrollo de chatbot el cual permita la gestión automática para la solicitud de pedidos a través de aplicaciones de mensajería en establecimientos comerciales como Gastrobares, de igual manera se realiza el análisis 4+1 planteando las vistas pertinentes para el desarrollo e integración de la arquitectura de forma correcta.

Propósito

La información correspondiente a casos de uso, requerimientos funcionales y requerimientos NO funcionales y la vista de despliegue se encuentran contenidos en el documento principal del proyecto.

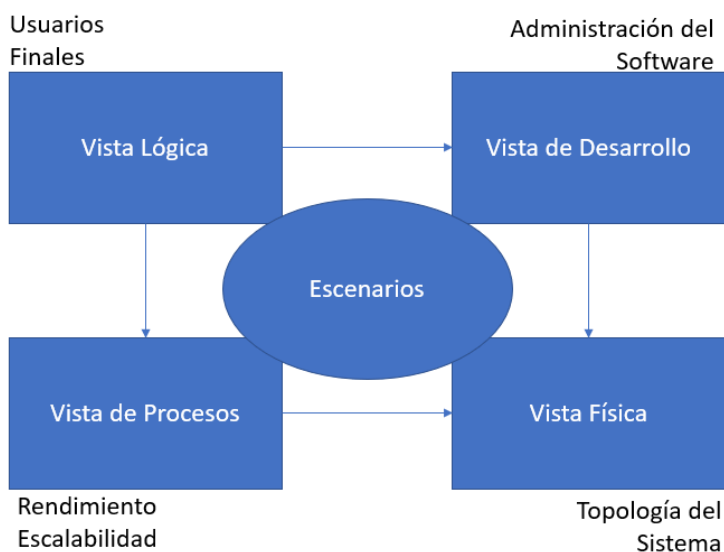
El presente documento fue elaborado con la finalidad de que se realice una visión y planteamiento general de la arquitectura que se va a realizar en cuanto al desarrollo de chatbot el cual permita la gestión automática para la solicitud de pedidos a través de aplicaciones de mensajería en establecimientos comerciales como Gastrobares, para ello se tendrán en cuenta el proceso realizado tanto con requerimientos funcionales y no funcionales y de los mismos atributos de calidad planteados dentro del documento de especificación de requerimientos de software.

A partir de este análisis se desarrollará el modelo de vistas 4+1, permitiendo el desarrollo de cada uno de los modelos de presentación de la solución.

A continuación, se detalla el modelo.

Ilustración 1

Modelo preliminar



Fuente: Diseño propio

Es importante tener en cuenta que este modelo está orientado a los usuarios finales e interesados, permite a todos los interesados, observar la información que requieran de una manera óptima.

Alcance

Este documento de arquitectura de software ha sido basado en el desarrollo de un chatbot el cual permita la gestión automática para la solicitud de pedidos a través de aplicaciones de

mensajería en establecimientos comerciales como Gastrobares y los puntos expuestos aquí fueron extraídas usando plantillas SAD.

Visión General

A continuación, se socializan los puntos mencionados en el alcance, según la plantilla de software de “software architecture document”

- Punto 2: Se describen las vistas y se explica porque y como se ejecutaron teniendo en cuenta los atributos de calidad y requerimientos
- Punto 3: Se explica y presenta las restricciones que poseerá el sistema
- Punto 4: Se describen los requerimientos arquitectónicamente significativos
- Punto 5: Se generan los casos de uso más importantes dentro de la arquitectura
- Punto 6: Se explica cómo se desplegará el sistema en los distintos ambientes
- Punto 7: Se realiza una descripción de capas y subsistemas de la aplicación

Representación Arquitectónica

A continuación, se describirán los aspectos generales de la arquitectura, Utilizando el modelo de vistas 4+1 y la convención de nomenclatura RUP.

- Vista Lógica: Esta vista va dirigida a Usuarios finales
 - ✓ Factores: Requerimientos funcionales, aquí se representará el caso de uso relevante para la construcción de la aplicación
 - ✓ Resultado: vista lógica

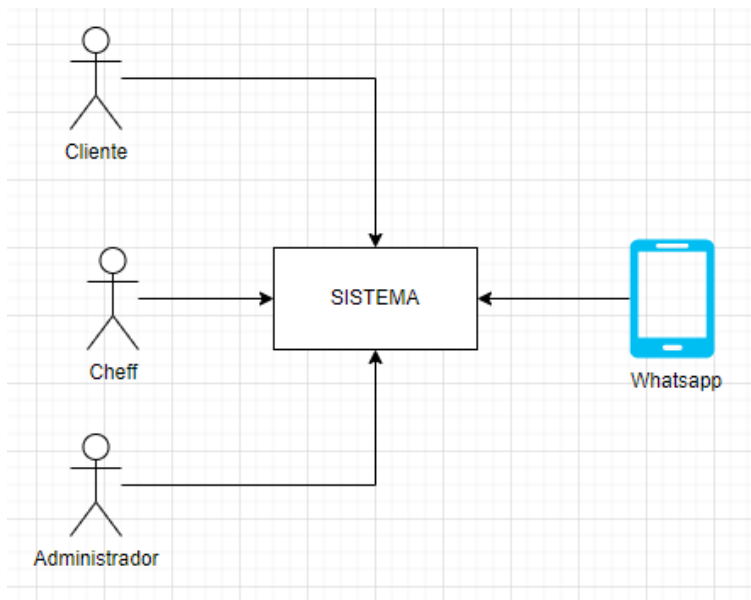
- Vista de procesos: Esta vista va dirigida a los integradores
 - ✓ Factores: Requerimientos no funcionales, se incluye todo lo que tiene que ver con el rendimiento de la aplicación (tiempos de respuesta) como también aspectos vitales de seguridad en el viaje de la información.
 - ✓ Resultado: Vista de procesos
- Vista de despliegue: Esta vista va dirigida al ingeniero Devops
 - ✓ Factores: Infraestructura de la aplicación, Relaciones entre componentes de hardware y software dependiendo de las necesidades y de los atributos de calidad
 - ✓ Resultado: Vista de despliegue
- Vista de Implementación: Esta vista va dirigida a Desarrolladores
 - ✓ Factores: Estructuración y organización de los módulos del sistema, describiendo los componentes y sus relaciones
 - ✓ Resultado: Vista de implementación
- Vista de casos de uso: Esta vista va dirigida a todos los interesados
 - ✓ Factores: Conocer las funcionalidades de manera global y cuáles de ellas serán implementadas dentro del sistema
 - ✓ Resultado: vista de casos de uso

Escenarios

A continuación, se describe el diagrama de Escenarios:

Ilustración 2

Diagrama de escenarios



Fuente: Diseño propio

Actor	Descripción
Cliente	Usuario del Gastrobar, será el encargado de solicitar pedidos, solicitar la cuenta e interactuar con las opciones de servicio del chatbot
Chef	Usuario que recibirá en pantalla los pedidos realizados por el cliente para que pueda comenzar con el proceso de preparación de alimentos
Administrador	Usuario que tendrá acceso a todas las características del sistema. Puede visualizar pedidos realizados, valores a cobrar y retroalimentaciones realizadas por el cliente. Así como administrar precios y productos
Whatsapp	Sistema externo que permitirá la comunicación entre el cliente y el comensal por medio del bot

Priorización

Hemos definido la priorización de los requerimientos en tres etapas:

1. Definir correctamente los criterios para la priorización de los requerimientos, catalogándolos en criterios de negocio como necesidades de usuarios, costo o técnicos como recursos existentes o factibilidad.

2. Determinar del orden de acuerdo con los criterios específicos para los diferentes participantes y stakeholders.

3. Utilizaremos el método conocido como MoSCoW que funciona más fácil para las partes interesadas y así poder calificar los requisitos como Must, Should, Could or Would

El acrónimo representa lo siguiente:

Must – obligatorio

Should – de alta prioridad

Could – Preferido, pero no necesario

Would – puede ser pospuesto y sugerido para futura ejecución

Al elaborar una lista de requerimientos con esta clasificación podríamos identificar cuáles son aquellos requerimientos que debemos atender de primero y así garantizar la disponibilidad del servicio y la continuidad del negocio.

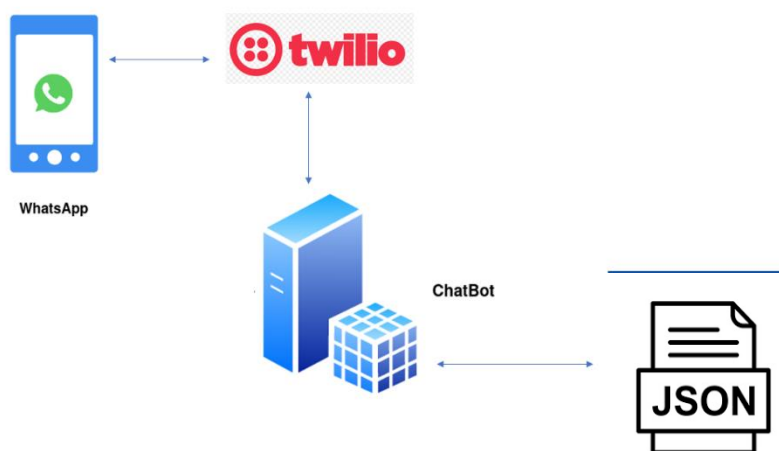
Visualización del Sistema

De acuerdo con la problemática y descripción presentada hemos establecido al sistema como una interconexión entre un servicio de mensajería un módulo funcional que contiene la

especificación de los flujos conversacionales del chatbot y la implementación de la red neuronal e IA que permiten la continuidad del negocio y de la solución definida al proyecto, además se define como parte fundamental del diseño la utilización de Json para el manejo de datos.

Ilustración 3

Visualización del sistema



Fuente: Diseño propio

WhatsApp: Aplicación de mensajería del comensal.

Twilio: Plataformas de comunicación en la nube que permite interacción entre clientes a través de los distintos canales: SMS, voz, video, correo electrónico, WhatsApp y más. En este proyecto enfocada al uso con WhatsApp.

ChatBot: En esta capa se ejecutará el modelo de negocio según lo planteado en la problemática, permitiendo realizar las funcionalidades de los requerimientos funcionales base y especializadas del sistema, esto se hará mediante la integración de solicitudes desde el servicio de mensajería y la capa de datos manejada con Json, sus funcionalidades estarían determinadas así:

- Configuración de las solicitudes para obtener las respuestas de los servicios de mensajería.
- Llamados a la data creada en la capa de datos para realizar el proceso de pedidos y solicitud de cuenta.
- Cifrado de datos para temas de transporte de la información y seguridad.

Stakeholders

ROL GENERAL	STAKEHOLDERS	OBSERVACIÓN
Patrocinador	Restaurante XY	Persona que permitirá poner en piloto la solución
Equipo del proyecto	Director de Proyecto (Scrum Máster)	Persona que genero la iniciativa de la aplicación
	Desarrollador	Persona técnica que construirá la aplicación
	Analista de Calidad	Persona que estará a cargo del proceso de calidad del proyecto y el producto
Usuarios /Clientes	Mesero	Persona que llevara el pedido y el cobro de cada mesa
	Chef	Persona que preparara el pedido
	Gerente restaurante	Persona que garantizara el trámite administrativo

Vista de Desarrollo

A continuación, se detallarán cada uno de los componentes que permitirán realizar las funcionalidades y ejecutar los servicios de la solución. Se puede apreciar una arquitectura multinivel o N-Tier que permite dividir la lógica por componentes físicos, permitiendo lograr mayor escalabilidad, simplicidad y hacer entendible el Código; de esta forma, se compone de módulos la solución, separando la lógica y de la persistencia, a continuación, se expone cada uno de los niveles de la arquitectura:

Nivel Vista: Debido a que este nivel corresponde al frontend es importante aclarar que para el chatbot esta capa no es necesaria ya que el usuario final interactuara únicamente con un contacto de WhatsApp que actuará como Bot y el cual será consumido a través de un componente de mensajería o actor externo llamado Twillio.

Nivel Lógica: será el nivel encargado de ejecutar las tareas del frontend (WhatsApp), es decir presentarle al usuario final las diferentes opciones de saludo, despedida, solicitud de pedidos, solicitud de cuenta, así como los diferentes mensajes automáticos que llegan a la aplicación, es en este nivel donde se ubicará el algoritmo de IA (Redes Neuronales) el cual se encargara de optimizar el uso del chatbot y humanizar la interacción con el mismo utilizando bibliotecas de Python como TensorFlow.

Este nivel también poseerá la lógica de respuestas multimedia (imágenes, pdf, audios y videos) para darle una experiencia única a el usuario final empleando bibliotecas de Python como SpeechRecognition.

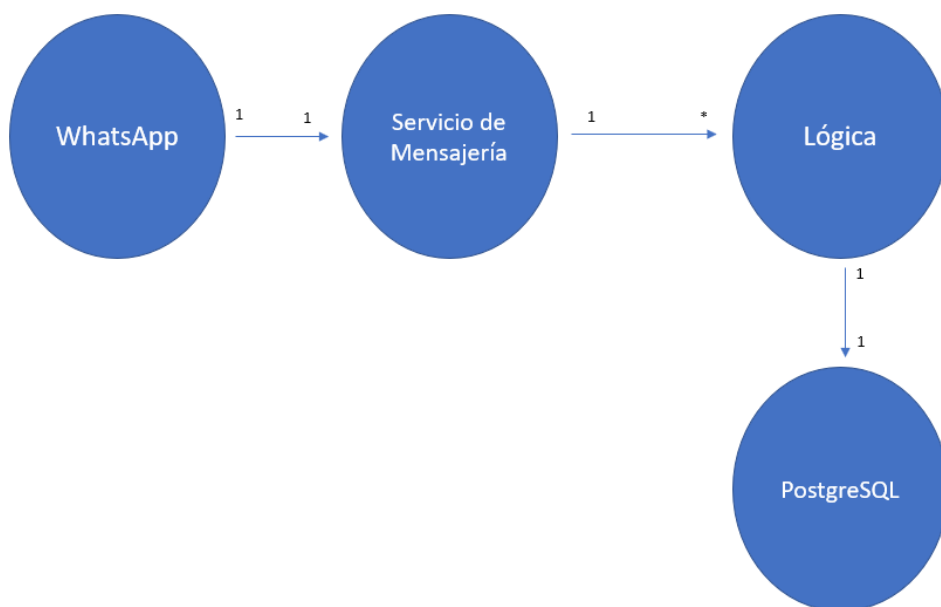
Este nivel permitirá la ejecución de la lógica de negocio contenida en el chatbot el cual permitirá que las solicitudes sean sobre productos que existen en stock.

Nivel de datos: corresponde al manejo de datos a través de Json permitiendo recopilar información que será utilizada en el backend.

Vista de Procesos

Ilustración 4

Vista de Procesos



Fuente: Diseño propio

Este diagrama define 4 procesos que deben estar ejecutándose para el correcto funcionamiento del negocio de la gestión automática para la solicitud de pedidos a través de aplicaciones de mensajería en establecimientos comerciales como Gastrobares así:

1. WhatsApp: Este es un proceso externo en el cual el comensal iniciara la conversación con el chatbot a través del escaneo de un código QR.
2. Servicio de Mensajería: Es un proceso externo que nos servirá de transporte para los mensajes recibidos y enviados.
3. Lógica: Proceso que se encontrara corriendo en una maquina servidor, el cual será múltiple ya que se emplea una cluster para suplir al un servidor cuando se haga mantenimiento o haya muchas solicitudes, este proceso contendrá toda la lógica del negocio respecto a la solicitud de pedidos y la liquidación de la cuenta cuando sea solicitada por el comensal.
4. Json: Es un formato ligero de intercambio de datos, de fácil lectura y escritura para los usuarios. También nos ayuda a realizar fácil análisis.

Patrones

Patron Broker:

Este patrón de arquitectura es utilizado para estructurar sistemas de software distribuidos con componentes desacoplados que interactúan por invocaciones de servicios remotos. Para la solución de este proyecto este patrón se acomoda bien a las necesidades especialmente por estas características:

Los componentes deben ser capaces de acceder a los servicios provistos de otros a través de invocaciones de servicios remotos sin conocer su ubicación.

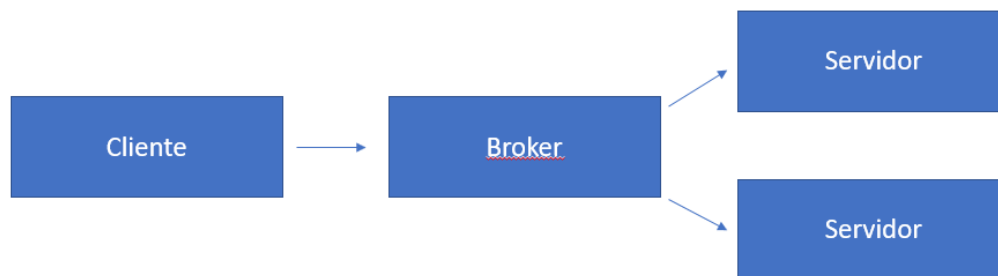
Se necesite intercambiar, añadir y quitar componentes en tiempo de ejecución.

La arquitectura debe esconder los detalles específicos de implementación del sistema de los usuarios de componentes y servicios.

Este patrón funciona básicamente así: los servidores se registran con el bróker y hacen sus servicios disponibles a los clientes a través de interfaces de métodos. Los clientes acceden a la funcionalidad enviándoles solicitudes a través del bróker. Este debe localizar el servidor apropiado, enviarle la solicitud y transmitir resultados y excepciones de vuelta al cliente.

Ilustración 5

Patron Broker



Fuente: (<https://aprendearquitecturasoftware.wordpress.com/2018/10/02/grupo-10-patron-de-arquitectura-software-broker/>)

CLIENTES

Son aplicaciones que acceden a los servicios de, al menos, un servidor. Para invocar los servicios, los clientes envían solicitudes al bróker. Cuando la operación se ha ejecutado, los clientes reciben respuestas o excepciones del bróker. La interacción entre clientes y servidores se basa en un modelo dinámico, lo cual significa que los servidores también pueden actuar de clientes.

SERVIDORES

Un servidor implementa objetos que exponen su funcionalidad a través de interfaces que están formados de operaciones y atributos. Las interfaces están disponibles a través de un lenguaje de definición de interfaz o un estándar binario. Hay dos tipos de servidores: los que ofrecen servicios comunes a muchos dominios de la aplicación y los servidores que implementan una funcionalidad específica para un dominio de aplicación particular.

BROKERS

Un bróker es un mensajero que se encarga de la transmisión de solicitudes de clientes a servidores y también de la transmisión de respuestas y excepciones de servidores a clientes. Lo hace de la siguiente manera: localiza al receptor de una solicitud basándose en un sistema de identificadores únicos. Ofrece APIs a clientes y servidores que incluyen operaciones para el registro de servidores y la invocación de métodos de servidores.

12.5 Documento de Gestión de servicios para el proyecto.

Introducción

La Gestión de servicios de TI tiene como objetivo distribuir de forma alienada la entrega de servicios de TI con las necesidades y objetivos estratégicos de las empresas. El objetivo principal de la Gestión de servicios es brindar las herramientas necesarias para ofrecer un servicio satisfactorio al usuario final.

La Gestión de servicios de TI se puede definir como una combinación de las políticas, procesos y métodos definidos para entregar productos y servicios de TI.

De acuerdo con estos lineamientos este documento pretende ofrecer un listado de aquellos documentos que son necesarios tener en cuenta e incluir en el desarrollo del proyecto denominado Gestión automática para la solicitud de pedidos a través de aplicaciones de mensajería en establecimientos comerciales como Gastrobares”.

Estrategia del servicio

Dentro del plan estratégico de servicios es importante generar una perspectiva de lo que se requiere desarrollando, elaborando y ejecutando planes de acción que nos permitirán tener un producto exitoso adoptando los patrones de acción necesarios y los cuales se detallarán en los documentos de este punto.

Es importante en nuestra estrategia adoptar los principios clave los cuales nos ayudaran a que el producto tenga en cuenta no solo aspectos técnicos sino también estratégicos del servicio como:

- La gestión de servicios como un activo estratégico.
- Los clientes internos y externos.
- Servicios internos y externos.
- Núcleo, habilitación y mejora de servicios.
- Valor; resultados empresariales, percepciones y preferencias.
- Funcionalidad (utilidad), el rendimiento (garantía) y el precio.
- Paquetes de servicios.
- Estrategia de suministros.
- Redes de valor.

El buen planteamiento de la estrategia se basa en una buena gestión de esta, por eso es fundamental generar una excelente estrategia, acompañada de una ejecución correcta, sin olvidar medir y evaluarla en todo momento

La gestión de la demanda es también vital para la estrategia del servicio ya que gracias a ella identificamos aspectos clave para llevar al mercado un producto que sea consumible, es por eso por lo que debemos realizar un análisis y codificación de patrones de la actividad del negocio que nos dé una perspectiva más acertada, también es importante desarrollar ofertas diferenciadas que generen valor al servicio y al producto

En nuestra estrategia es muy conveniente realizar trámites que permitan tener buenas relaciones de negocios, por tal motivo el manejo de peticiones y quejas nos permitirán mejorar nuestro servicio y producto y por consiguiente identificar nuevas oportunidades.

No debemos olvidar realizar una buena gestión de portafolio de servicios que nos permita definir, analizar y probar los servicios en desarrollo para mejorar nuestro catálogo.

Las declaraciones de misión y visión.

Gestionar de manera automática la solicitud de pedidos y valor de la cuenta directamente a la cocina y caja respectivamente de los comensales en los establecimientos comerciales como Gastrobares, usando aplicaciones de mensajería con respuestas automáticas implementando en ellas Inteligencia Artificial.

Posicionarnos en el mercado de creaciones de Chatbot diferenciándonos de la competencia haciendo de estos Chatbots más amigables, accesibles y hasta divertidos implementando inteligencia artificial en cada uno de los diferentes flujos.

Estrategias y tácticas

Prestar un servicio de calidad depende de del conocimiento profundo que se tenga de los potenciales clientes, por tanto, definimos la estrategia para ofrecer nuestro servicio en los siguientes pasos:

- Publicidad efectiva off y online.
- Emplear redes sociales para hacer marca.
- Compartir el conocimiento y la experiencia en la construcción de Chatbot a través de Webinar.
- Interactuar personalmente con los clientes.
- Realizar sesiones de entrenamientos de manera continua.

Definiciones de los servicios, la clasificación y visualización.

Con esta solución de chatbot para Gastrobares basado en WhatsApp se ofrece una interacción entre los clientes de estos establecimientos comerciales y los administradores de este para generar y establecer un mejor nexo con ellos de una manera conversacional, que generará una mayor empatía y brindará una experiencia diferente, agradable y amena a la hora de realizar las solicitudes de pedido o de cuenta a pagar.

Esta solución va dirigida especialmente a establecimientos comerciales como Gastrobares y puede ser ampliada a una red de restaurantes de cualquier cadena comercial. Inicialmente estará dirigido a establecimientos comerciales en la ciudad de Bogotá, a medida que se vaya conociendo el proyecto, éste puede expandirse a diferentes ciudades.

Los Casos de Negocio

Gestionar de manera automática la solicitud de pedidos y valor de la cuenta directamente a la cocina y caja respectivamente de los comensales en los establecimientos comerciales como Gastrobares, usando aplicaciones de mensajería con respuestas automáticas implementando en ellas Inteligencia Artificial.

Tabla 1

Objetivos e indicadores de éxito

Objetivo	Métrica	Indicador de éxito
<p>Entregar una implementación en la cual el 60% del flujo conversacional con el Chatbot sea con Inteligencia Artificial.</p> <p>Afianzar que los tiempos de respuesta en la interacción con el Chatbot no supere los 5 segundos a pesar de la implementación de componentes de Inteligencia Artificial.</p>	<p>Utilizando herramientas de análisis de código se puede establecer el porcentaje de código que será implementado con Inteligencia artificial.</p>	<p>Lograr un porcentaje mayor al 60% de cobertura.</p>
<p>Humanizar los flujos conversaciones del Chatbot a través de componentes multimedia (audio/imágenes) en un 70% de los flujos.</p>	<p>Utilizamos la misma métrica del objetivo anterior.</p>	<p>Lograr un porcentaje mayor al 70% de cobertura.</p>

Fuente: Diseño Propio

Análisis de Impacto a Negocio (BIA)

Se han identificado tres clases de impactos en los cuales este proyecto puede afectar al sector de Gastrobares tanto para los clientes como para los dueños de estos establecimientos comerciales:

Impacto Social

Un proyecto de estas características impactaría significativamente al recurso humano ya que se disminuiría la interacción entre clientes y meseros de los diferentes establecimientos. El hecho de implementar los diferentes protocolos de bioseguridad decretados por el Gobierno Nacional ha generado un cambio estructural respecto al servicio prestado en estos negocios por lo que este proyecto puede ayudar a que estos protocolos se puedan implementar haciendo uso de esta tecnología y así mantener una conversación con el Chatbot.

Impacto tecnológico

Aunque con este proyecto pretende usar las herramientas de comunicación más utilizadas por diferentes usuarios en el mundo como es el WhatsApp y no necesite de descargas adicionales de aplicaciones, el restaurante deberá ser consciente de adquirir equipos tecnológicos que permitan que esta solución tenga éxito ya que gracias a la interacción entre el cliente y el Chatbot esta información será mostrada tanto al chef como el gerente a través de dispositivos periféricos.

Impacto económico

Reemplazo en costos administrativos por costos operativos: Debido a la disminución de personal ya que la interacción entre clientes y meseros es mínima esta será sustituida por los

costos que generará el servicio de mensajería de Twilio que es mínimo en comparación con la carga prestacional y/o administrativa del establecimiento

Diseño de Servicio

En el diseño de servicio es necesario generar una planeación que permita analizar el diseño y si es necesario tener algunos puntos de mejora de este revisándolo y modificándolo si es necesario. Es importante tener soluciones alternativas que permitan mejorarlo.

Para tener un diseño del servicio integral es necesario tener en cuenta aspectos tales como:

- Diseño de soluciones para servicios nuevos o modificarlos.
- Diseño de sistemas de información y herramientas de gestión, SPM.
- Diseño de Arquitectura (TI y Gestión)
- Diseño de procesos.
- Diseño de métricas.
- Diseño equilibrado
- Diseño de servicios de TI frente a cambios en el negocio

Esto nos permitirá involucrar y hacer parte activa del diseño de servicio a actores muy importantes como las personas, procesos, productos, proveedores que son vitales en el adecuado diseño.

La gestión de catálogos de servicio, niveles de servicio, proveedores, continuidad del servicio y seguridad de la información son fundamentales y hacen parte en todo momento del

óptimo diseño ya que definen una excelente interacción con todos los actores, sin descuidar el servicio ni algunos aspectos legales que hacen parte de él.

Se requiere generar una coordinación de servicios eficiente tanto a nivel general como para mejorar el diseño de los servicios para cada diseño y para esto definir los aspectos clave en ellos.

Políticas y planes de diseño de servicios

Política

- El diseño de servicio deberá ejecutarse en los formatos asignados para tal fin, los cuales podrán ser descargados en la página web institucional.
- Todas las solicitudes de diseño deberán hacerse llegar en el formato antes mencionado al correo: dialejorod@gmail.com cc: sayaroga@gmail.com cuyo asunto deberá enviarse así: Formato Diseño de Servicio 001 (Nombre del Formato) (Área) (Fecha –Abreviado el Año-) Con un mínimo de 3 a 5 días hábiles para proyectos en curso y un mínimo de 15 a 20 días hábiles para proyectos nuevos que involucren a un proveedor externo. Esta solicitud deberá venir completa, con correcta ortografía y redacción.
- Las solicitudes de diseño serán atendidas y se brindarán tiempos de entrega de acuerdo con los siguientes parámetros:
 1. Fecha de solicitud.
 2. Relevancia del proyecto.
 3. Complejidad u horas empleadas 4.-Volumen de trabajo asignado.
- En caso de requerir el servicio de desarrollo para el diseño, favor de especificar.

- La gerencia no se hace responsable del contenido proporcionado (ortografía, redacción, origen y contenido de la información).
- Las propuestas de diseño serán enviadas al correo donde se realizó la solicitud, para su revisión, el límite de modificación al diseño será tres, una vez validado se pedirá su firma de autorización.
- Para los proyectos que involucran producción con proveedor externo, el área solicitante deberá realizar su respectiva requisición, cotización y le dará seguimiento con el área de compras.
- Es importante mencionar que cualquier interacción de chatbot que tenga logo formal y como parte del diseño del servicio se desee ser cambiada deberá ser primero autorizada y generada por el especialista.

Plan

Para implementar el plan de diseño de servicio tuvimos en cuenta los siguientes aspectos que harán tener un servicio y producto de calidad:

- Fijar Objetivos y expectativas que se tienen a nivel de servicio, tanto a corto, mediano y largo plazo, sabiendo cuales son las fortalezas y valores para poder tener un servicio y un producto de calidad.
- Detectar puntos de mejora y oportunidades en base a la información obtenida. Explorar nuevas ideas en base a ello y trabajar en el valor del proyecto, todo aquello que pueda marcar la diferencia con la competencia.

- Realizar pruebas del servicio con un pequeño grupo de personas que cumplan con el perfil del cliente. Este ejercicio es ideal para recibir retroalimentación de las cuales se obtendrá posibles mejoras.
- Realizar una lista de tareas con acciones claras, que estarán definidas según los objetivos y lo que el cliente espera.

Asegurar que la estrategia sea clara y no ejecutarla a ciegas, sino que todos y cada uno de los pasos que se dan siguen un plan claro que se ha establecido.

Transición del servicio

En la transición del servicio es indispensable generar un proceso de planeación y preparación teniendo en cuenta el plan de versión, construcción, pruebas y despliegue y así realizar un buen cierre del servicio.

Una buena transición de servicio se lleva a cabo a partir de los siguientes principios clave:

- Políticas para la transmisión del servicio.
- Administrar todos los cambios a través de la transición de servicios.
- Marco común y las normas.
- Reutilización.
- Dirección de la transición.
- Relaciones con interesados.
- Los controles efectos y disciplinas
- Transferencia de conocimientos.
- Paquetes de versión.

- Planificación y gestión de recursos.
- La participación temprana.
- Gestión de la calidad y la mejora.
- Big bang vs Gradual.
- Push and pull.
- Automatización vs Manual.
- Datos - Información - Conocimiento - sabiduría

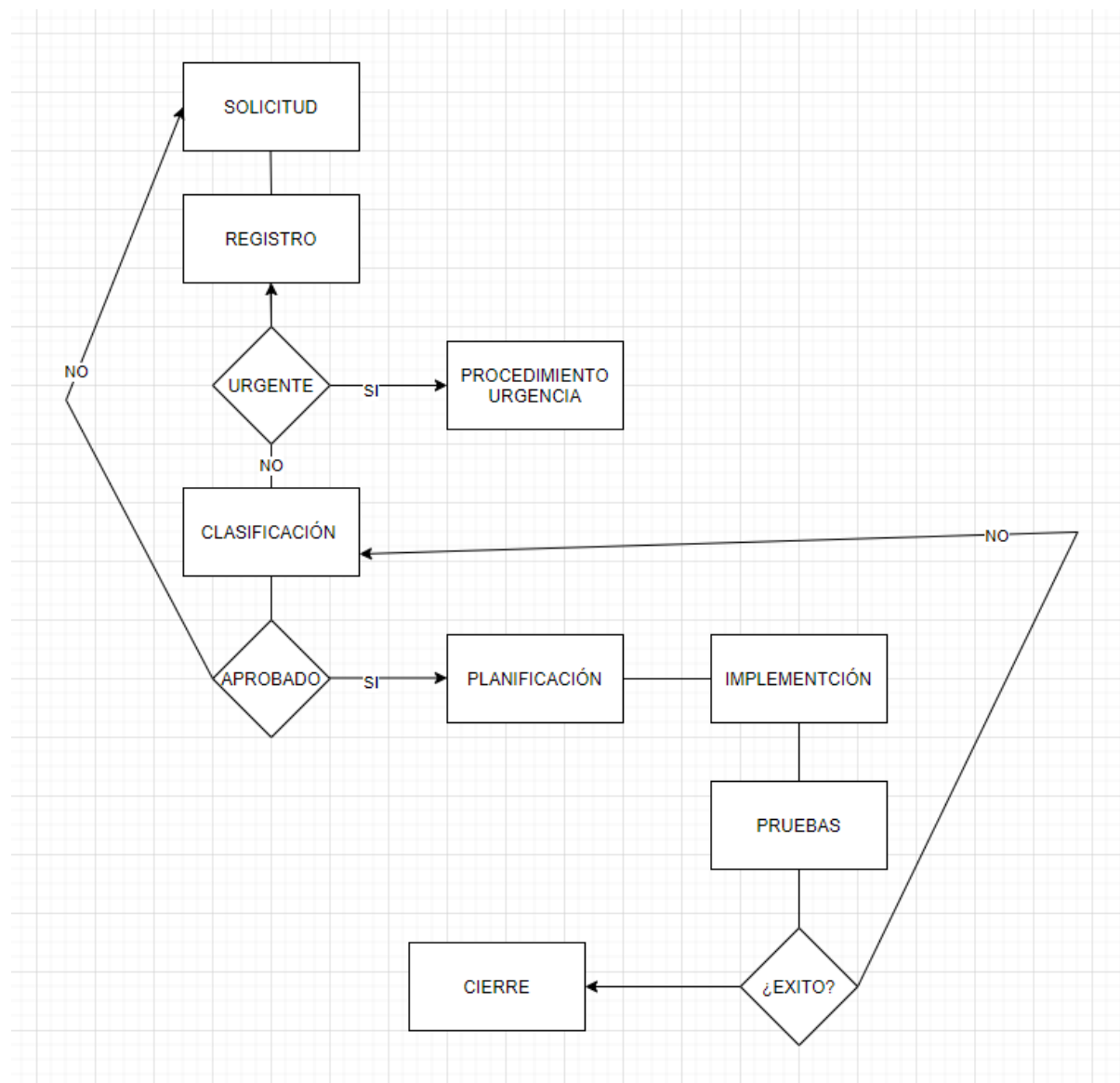
Sin embargo, es muy importante dentro de la transición todo lo que tiene que ver con pruebas y validaciones para tener servicios y productos de calidad que garanticen su buen funcionamiento de cara al cliente final.

Como es normal en el ciclo de vida del servicio, los aspectos vinculados a la gestión son fundamentales en este punto es por esto por lo que es indispensable hacer los controles necesarios en la configuración y activos del servicio, tener en cuenta en esta gestión las versiones y despliegues y definir una estrategia para no perder el conocimiento

Modelos de cambio.

Ilustración 1

Modelo de cambio 1



Fuente: Diseño Propio

Modelos de configuración.

Todos, tanto los componentes de los servicios TI como los servicios que éstos ofrecen, constituyen diferentes elementos de configuración. A continuación, mencionaremos los relevantes de este proyecto:

- Dispositivos de hardware como PCs, impresoras, routers, monitores, etc. así como sus componentes: tarjetas de red, teclados, lectores de CDs
- Software: sistemas operativos, aplicaciones, protocolos de red.
- Documentación: manuales, acuerdos de niveles de servicio,

En resumen, estos componentes han de ser gestionados por el equipo de infraestructura el cual tendrá la responsabilidad de garantizar dicha configuración.

Dichos elementos son seleccionados por el equipo de infraestructura y aprobados por la gerencia y dirección del proyecto para después ser agrupadas, clasificados e identificados de manera que puedan ser gestionados por el equipo designado para tal labor

El equipo de infraestructura tendrá una base de datos de gestión de la configuración la cual será usada para guardar los registros de configuración a lo largo de su ciclo de vida. El sistema de gestión de la configuración mantiene una o más bases de datos de la gestión de la configuración, cada una de estas bases de datos contiene los atributos de los elementos de configuración y sus relaciones con otros elementos.

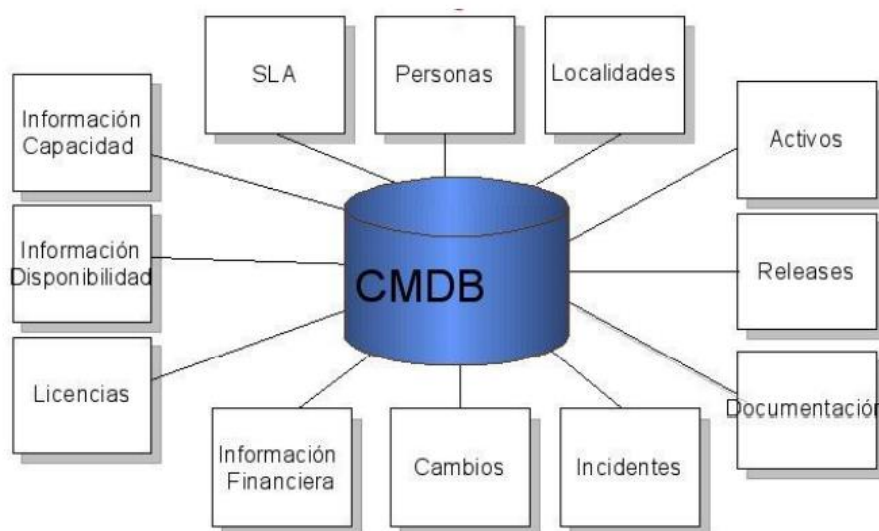
Los procesos automatizados para cargar y actualizar la base de datos de la gestión de la configuración deberán ser desarrollados, en lo posible, para reducir errores y gestionar costos.

Esta base de datos debe incluir:

- Información detallada de cada elemento de configuración.
- Interrelaciones entre los diferentes elementos de configuración, como, por ejemplo, relaciones "padre-hijo" o interdependencias tanto lógicas como físicas

Ilustración2

CMDB



Fuente: Diseño Propio

Estrategia de gestión de conocimiento

Identificación del conocimiento

Se identificará dentro de cada área las personas y el tipo de conocimiento que cada una de ellas tenga, posteriormente se almacenará esta información en una base de datos agrupándola por temas clave del nicho de mercado en el que estamos trabajando.

Adquisición del conocimiento

Una vez identificado el conocimiento en la organización, este crece y se multiplica en la medida en que se utiliza. Esto exige generar un constante proceso de transformación, y trabajar intensamente para generar células capaces de distribuir esta información a otras células

Igualmente, y tomando en cuenta que el conocimiento se expresa por medio de la información y que esta debe registrarse en documentos que respalden el accionar de la organización, se apunta que todo sistema que gestiona conocimiento debe disponer para el desarrollo del proceso de adquisición efectiva de los sistemas de información y de gestión documental.

Desarrollo del conocimiento

Ya que es posible que la organización no cuente con determinado conocimiento, se deben crear condiciones e invertir para su desarrollo. Este proceso de creación o desarrollo del conocimiento no es más que un proceso de desarrollo de las competencias y habilidades de los individuos que pertenecen a la organización, es un proceso donde se propicia el establecimiento de un ambiente que favorezca el surgimiento de nuevas ideas para fomentar la innovación.

Distribución del conocimiento (compartir)

El conocimiento organizacional puede proceder de fuentes internas, propias de la organización, o externas, cuando se adquiere de otras. Si se encuentran localizados e identificados los activos del conocimiento en la organización, entonces es posible compartir y distribuir el conocimiento, haciendo sesiones con las personas expertas y generando equipo multifuncionales.

Si el conocimiento no se obtiene de fuentes internas, será necesario generar sesiones con agentes externos que fortalezcan la capacidad de nuestro equipo de trabajo

Uso del conocimiento

En el ciclo de los procesos estratégicos de la gestión del conocimiento, el uso del conocimiento se ubica casi al final; sin embargo, esta ubicación es relativa, debido a que los procesos de identificación, adquisición, desarrollo y distribución del conocimiento siempre se encuentran en consonancia con las necesidades de los usuarios. Por eso, es necesario considerar un sistema de gestión de información que facilite información actualizada sobre las necesidades de los usuarios con vistas a lograr una eficiente gestión del conocimiento.

Medición del conocimiento

Se evaluará constantemente si se cumplen o no los propósitos del conocimiento en la empresa. Para esto se aplicarán diferentes técnicas de evaluación, como sesiones prácticas o evaluaciones individuales

Operación del Servicio

El seguimiento y control de los servicios es el corazón de la etapa de operación del servicio, por lo que es muy importante gestionar los servicios, componentes y actividades para así optimizar el servicio y tener un producto y servicio de calidad.

En la operación del servicio es clave adoptar algunos principios los cuales nos ayudaran a que la operación fluya con normalidad, estos principios serían:

- Servicios de TI frente a componentes tecnológicos.

- Estabilidad frente a la capacidad de respuesta.
- Calidad vs Costo.
- Reactiva v Proactiva.
- Participación en otras etapas del ciclo de vida de servicio.
- Salud operacional.
- Comunicación.
- Documentación.

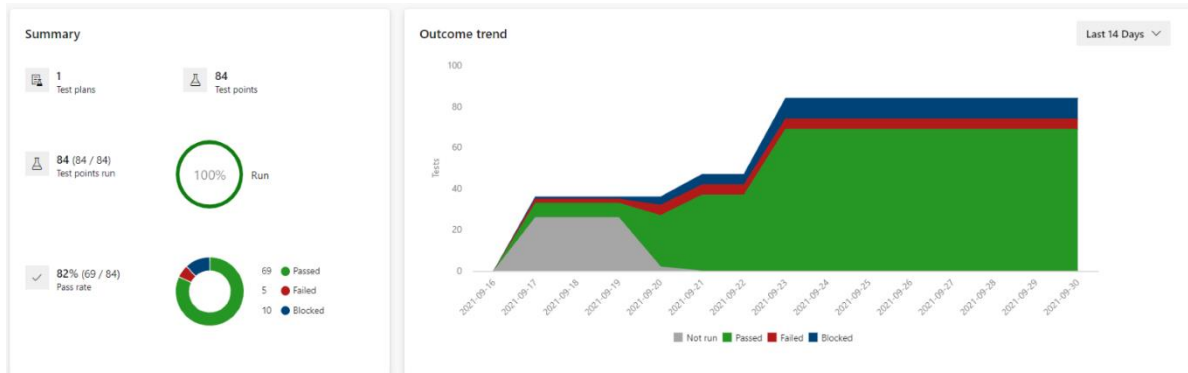
Hay que tener en cuenta que en el ciclo de vida del servicio en el punto de operaciones se encuentra la mayor parte de incidencias y problemas, es por esto por lo que se debe garantizar la gestión de estas haciendo un proceso adecuado.

Políticas de gestión de incidentes planes e informes

- Los incidentes serán gestionados en la herramienta definida para tal fin con las características acordadas en todas las áreas que hacen parte del servicio.
- Todas las incidencias deberán hacerse llegar en la herramienta antes mencionada y ser asignadas para tramite al correo: dialejorod@gmail.com cc: sayaroga@gmail.com.
- Los incidentes se tramitarán en un estado inicial de nuevo, una vez sea tramitado por el área encargado el estado cambiara a ejecutado, y cuando es verificada la solución el estado terminara en solucionado.
- Se realizará un dashboard pertinente al análisis que se realice de la cantidad de incidentes que se gestionen.

Ilustración 3

Diseño Dashboard



Fuente: Diseño Propio

Modelos de solicitud.

Tabla 2

Descripción tabla de solicitudes

ACTIVIDAD	DESCRIPCIÓN	TAREAS
SOLICITUD DEL CLIENTE	El cliente manifiesta una necesidad y realiza una solicitud mediante (RFP, PROPUESTA, POC, DEMO, ASSESSMENT, PREVENTA)	1. Recepción de la necesidad
INGRESO REQUERIMIENTO A HERRAMIENTA	El Analista encargado de la cuenta realiza el ingreso del requerimiento en las herramientas corporativas	1. Subir el requerimiento a las Herramientas
		2. Actualizar la información del cliente
		3. Enviar solicitud a dirección de operaciones y al integrador.
APROBACION DEL REQUERIMIENTO	La dirección de operaciones y/o el integrador realizan l aprobación del requerimiento para ser pasado al equipo técnico	1. Realizar aprobación, firma del requerimiento
		2. Seleccionar el checklist de servicios para pasarla al equipo técnico
		3. Enviar notificación a las diferentes áreas involucradas (Desarrollo, QA, DevOps,Procees Factory Arquitectura, entre otros)
EJECUCIÓN DEL REQUERIMIENTO	El área de Desarrollo esta encargada de poner en marcha y ejecutar el requerimiento	1. Realizar el detalle de las tareas técnicas que se pretenden empezar a ejecutar por el equipo 2.Crear los sprints correspondientes y realizar las estimaciones necesarias

REALIZAR ENTREGABLE	Las áreas involucradas (QA,dev,devOps y Arquitectura) realizaran el entregable y lo llevaran a un ambiente productivo	1.Programar reunión de entregable y socialización funcional. 2.Recibir feedback de las areas que hacen parte de la entrega 3.Realizar la firma de manera formal.
---------------------	--	--

Diseño Propio

Diseño del Servicio

La mejora continua del servicio tiene un enfoque de monitorear el ciclo de vida del servicio y a través de las métricas de este realizar planes de acción para mejorarlo, es por esto por lo que debemos seguir algunos principios clave de medición y gestión para ejecutar cambios específicos con los que el servicio y el producto mejore día a día.

ITIL nos habla en este punto de la mejora en 7 pasos los cuales nos hablan de analítica de datos, por tal razón es muy importante la toma de decisiones soportada en datos que nos permitan plantear estrategias y a responder 6 preguntas clave para dar la mejora que requiere el servicio:

1. ¿Cuál es la visión?

Posicionarnos en el mercado de creaciones de Chatbot diferenciándonos de la competencia haciendo de estos Chatbots más amigables, accesibles y hasta divertidos implementando inteligencia artificial en cada uno de los diferentes flujos.

2. ¿Dónde estamos ahora?

Estamos en el proceso de desarrollo para la solución del Chatbot propuesto, en la etapa de planeación y diseño.

3. ¿Dónde queremos estar?

Queremos tener un producto total mente construido q cumpla las expectativas de nuestros futuros clientes usando tecnologías de nuevas tendencias como la Inteligencia Artificial y Big Data.

4. ¿Cómo llegamos allí?

Formando un equipo de desarrollo con los conocimientos adecuados para construir una aplicación y teniendo un buen equipo de ventas para impulsar la comercialización de nuestro producto y servicio en los gastrobares.

5. ¿Hemos conseguido llegar?

Aún no, estamos en el proceso inicial como parte de la formación en la especialización de Ingeniera de Software.

6. ¿Cómo mantenemos el impulso?

Actualizándonos todos los días en la parte tecnología Y fortaleciendo los conocimientos y generando negocios que nos permitan tener un retorno de la inversión.

Para la mejora continua del servicio existen técnicas comunes enfocadas en la medición y análisis de métricas soportando servicios de información, matrices DOFA con las que se pueda buscar un retorno de inversión.

Factores Críticos de Éxito (CSF)

A continuación, definimos los factores críticos de éxito:

1. Disponibilidad 7*24 del chatbot para que el cliente pueda siempre tener la opción de consultar o recibir las alertas del caso
2. Velocidad de respuesta ante las preguntas generadas por el cliente al Chatbot esta debe ser inferior a un tiempo de espera de 2 segundos.
3. Menu actualizado con precios competitivos en el mercado
4. Respuestas lógicas y coherentes ante las preguntas realizadas por el cliente final en la interacción con el chatbot.

Indicadores Clave de Rendimiento (KPI), métricas y logros.

Se identificaron los siguientes indicadores de rendimiento para obtener métricas que permitan generar una mejora continua del servicio así:

Tabla 3

Indicadores de Rendimiento

INDICADOR	DESCRIPCIÓN
Cantidad de quejas de clientes	Cantidad de quejas recibidas de los clientes

Cantidad de quejas de clientes aceptadas	Cantidad de quejas recibidas de los clientes que fueron aceptadas como justificadas
Cantidad de encuestas de satisfacción de clientes	Cantidad de encuestas de satisfacción de clientes formales realizadas durante el periodo del informe
Porcentaje de Cuestionarios Encuesta	Porcentaje de Cuestionarios Encuesta, en relación a la cantidad total enviada
Cantidad de Evaluaciones de Servicios	Cantidad de Evaluaciones de Servicios realizadas durante el periodo del informe
Cantidad de debilidades identificadas	Cantidad de puntos débiles identificados durante la Evaluación de Servicio, para ser tratados mediante iniciativas de mejoras
Cantidad de Comparativas de Procesos, Evaluaciones de Madurez, y Auditorías	Cantidad de Comparativas de Procesos formales, Evaluaciones de Madurez, y Auditorías realizadas durante el periodo del informe
Cantidad de Evaluaciones de Procesos	Cantidad de Evaluaciones de Procesos formales realizadas
Cantidad de debilidades identificadas	Cantidad de puntos débiles identificados durante la Evaluación de Procesos, para ser tratados mediante iniciativas de mejoras
Cantidad de Iniciativas de CSI	Cantidad de Iniciativas de CSI, resultando de los puntos débiles identificados durante la Evaluación de Servicios y Procesos
Cantidad de Iniciativas de CSI completadas	Cantidad de Iniciativas de CSI que fueron completadas durante el periodo del informe

Diseño Propio

Cuadros de mando integral.

Tabla 4

Cuadro de mando

PERSPECTIVA	OBJETIVO	INDICADOR	METAS	INICIATIVA
FINANCIERA	Crecimiento en los ingresos por servicios actuales	Crecimiento anual de ingresos	25%	
	Crecimiento en los ingresos por servicios nuevos	Crecimiento anual por nuevos servicios	20%	
	Mejora de Productividad	Mejora de productividad para nuevos servicios	30%	
CLIENTES	Fidelización de clientes	Clientes nuevos fidelizados	90%	
	Satisfacción de clientes	Encuesta de satisfacción de servicio	100%	
PROCESOS INTERNOS	Asegurar la calidad del producto	Incidencias técnicas para el producto	0%	
	Asegurar la calidad del servicio	Incidencias en calidad del servicio	0%	
	Mejorar la calidad de respuesta	Tiempo de respuesta de preguntas al cliente	500 milisegundos	
APRENDIZAJE Y CRECIMIENTO	Aumentar la formación	Actualización en nuevas tendencias tecnológicas	15 horas	

	Retención de personal clave	Retención de personal clave	95%	
--	-----------------------------	-----------------------------	-----	--

Diseño Propio

Planes de Mejora de Servicio (SIP)

Debido a que el servicio en este proyecto está directamente relacionado con la tecnología ya que la cara al cliente es un Chatbot, es necesario realizar monitoreos continuos teniendo en cuenta lo siguiente:

1. Monitoreo del Chatbot en tiempo real, saber cómo es la interacción de los clientes con el Bot.
2. Existirá una persona a cargo de tomar decisiones soportadas en métricas que será la propietaria del proceso y del servicio
3. Propietario de la iniciativa o cambio que genere mejora. Estos cambios se realizan en un 90% a las respuestas que genera el Chatbot y así lograr más empatía con los clientes finales.
4. Todas las iniciativas de mejora deben ser aprobadas por la dirección de TI.
5. Estas iniciativas se generarán con los formatos y descripciones correspondientes.
6. Origen de la medida (Soporte de analítica de datos para concluir que hay necesidad de generar un cambio en el servicio)
7. Monitoreo constante del caso de negocio
 1. Resultado generado de la iniciativa
 2. Estimación del costo (tanto a nivel tecnológico como del servicio)

3. Resultado deseado de la iniciativa para realizar comparación entre el generado

8. Calendario de implementación, estado actual y métricas

1. Fecha prevista
2. Estado actual
3. Métricas diarias

Conclusiones

Hemos concluido con la elaboración de este documento que la organización independiente del tamaño del negocio debe estar involucrada en la gestión de servicios de TI para poder garantizar que los incidentes, las solicitudes de servicio, los problemas, los cambios y los activos de TI, además de otros aspectos importantes puedan ser gestionados de forma simplificada y clara.

Realizando la implementación de estos procesos de gestión de servicios podemos garantizar varias cosas importantes que son relevantes en cualquier tipo de organización, estas podrían estar en la siguiente lista de beneficios que nos brinda la gestión de servicios:

- Menores costos para las operaciones y mayores retornos de inversión.
- Roles y responsabilidades bien definidos
- Expectativas claras sobre los niveles de servicio y la disponibilidad del servicio y mayor transparencia en estos procesos.
- Interrupciones del servicio mínimas.